

Compaq Computer Corporation

TPC Benchmark™ C
Full Disclosure Report
for
Proliant 7000-6/450-2S
using
Microsoft SQL Server 7.0 Enterprise Edition
and
Window NT 4.0 Enterprise Edition (SP4)

First Edition
February-99

Compaq Computer Corporation (Compaq) believes that the information in this document is accurate as of the publication date. The information in this document is subject to change without notice. Compaq assumes no responsibility for any errors that may appear in this document. The pricing information in this document is believed to accurately reflect the current prices as of the publication date. However, Compaq provides no warranty of the pricing information in this document.

Benchmark results are highly dependent upon workload, specific application requirements, and system design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, TPC Benchmark C should not be used as a substitute for a specific customer application benchmark when critical capacity planning and/or product evaluation decisions are contemplated.

All performance data contained in this report were obtained in a rigorously controlled environment. Results obtained in other operating environments may vary significantly. Compaq does not warrant or represent that a user can or will achieve similar performance expressed in transactions per minute (tpmC) or normalized price/performance (\$/tpmC). No warranty of system performance or price/performance is expressed or implied in this report.

Copyright 1999 Compaq Computer Corporation.

All rights reserved. Permission is hereby granted to reproduce this document in whole or in part provided the copyright notice printed above is set forth in full text or on the title page of each item reproduced.

Printed in U.S.A., 1999

Compaq, NonStop, Proliant 7000, and Proliant are registered trademarks of Compaq Computer Corporation.

Microsoft, Window NT and SQL Server for Window NT are registered trademarks of Microsoft Corporation.

Pentium II is a registered trademark of Intel.

Performix is a trademark of Pure Atria, Inc.

TPC Benchmark is a trademark of the Transaction Processing Performance Council.

Other product names mentioned in this document may be trademarks and/or registered trademarks of their respective companies.

Table of Contents

TABLE OF CONTENTS	3
PREFACE	5
TPC BENCHMARK C OVERVIEW	5
ABSTRACT	6
OVERVIEW	6
TPC BENCHMARK C METRICS	6
STANDARD AND EXECUTIVE SUMMARY STATEMENTS	6
AUDITOR	6
GENERAL ITEMS	10
TEST SPONSOR	10
APPLICATION CODE AND DEFINITION STATEMENTS	10
PARAMETER SETTINGS	10
CONFIGURATION ITEMS	10
CLAUSE 1 RELATED ITEMS	13
TABLE DEFINITIONS	13
PHYSICAL ORGANIZATION OF DATABASE	13
<i>Benchmarked Configuration:</i>	13
PRICED CONFIGURATION VS. MEASURED CONFIGURATION:	14
INSERT AND DELETE OPERATIONS	14
PARTITIONING	14
REPLICATION, DUPLICATION OR ADDITIONS	14
CLAUSE 2 RELATED ITEMS	15
RANDOM NUMBER GENERATION	15
INPUT/OUTPUT SCREEN LAYOUT	15
PRICED TERMINAL FEATURE VERIFICATION	15
PRESENTATION MANAGER OR INTELLIGENT TERMINAL	15
TRANSACTION STATISTICS	15
QUEUING MECHANISM	16
CLAUSE 3 RELATED ITEMS	17
TRANSACTION SYSTEM PROPERTIES (ACID)	17
ATOMICITY	17
<i>Completed Transactions:</i>	17
<i>Aborted Transactions:</i>	17
CONSISTENCY	17
ISOLATION	17
DURABILITY	18
<i>Durable Media Failure:</i>	18
<i>Instantaneous Interruption and Loss of Memory:</i>	18
CLAUSE 4 RELATED ITEMS	21
INITIAL CARDINALITY OF TABLES	21
DATABASE LAYOUT	21
TYPE OF DATABASE	21

DATABASE MAPPING.....	22
180 DAY SPACE	22
CLAUSE 5 RELATED ITEMS.....	23
THROUGHPUT	23
KEYING AND THINK TIMES	23
RESPONSE TIME FREQUENCY DISTRIBUTION CURVES AND OTHER GRAPHS	24
FIGURE 10. THROUGHPUT VS. TIME DISTRIBUTION.....	28
STEADY STATE DETERMINATION	29
WORK PERFORMED DURING STEADY STATE	29
REPRODUCIBILITY	29
MEASUREMENT PERIOD DURATION.....	30
REGULATION OF TRANSACTION MIX	30
TRANSACTION STATISTICS.....	30
CHECKPOINT COUNT AND LOCATION	30
CLAUSE 6 RELATED ITEMS.....	31
RTE DESCRIPTIONS.....	31
EMULATED COMPONENTS	31
FUNCTIONAL DIAGRAMS	31
NETWORKS	31
OPERATOR INTERVENTION.....	31
CLAUSE 7 RELATED ITEMS.....	32
SYSTEM PRICING.....	32
AVAILABILITY, THROUGHPUT, AND PRICE PERFORMANCE.....	32
COUNTRY SPECIFIC PRICING.....	32
USAGE PRICING.....	32
CLAUSE 9 RELATED ITEMS.....	33
AUDITOR'S REPORT	33
AVAILABILITY OF THE FULL DISCLOSURE REPORT	33

Preface

The TPC Benchmark C was developed by the Transaction Processing Performance Council (TPC). The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry. This full disclosure report is based on the TPC Benchmark C Standard Specifications Version 3.4, released August 25, 1998.

TPC Benchmark C Overview

The TPC describes this benchmark in Clause 0.1 of the specifications as follows:

TPC Benchmark C is an On Line Transaction Processing (OLTP) workload. It is a mixture of read-only and update intensive transactions that simulate the activities found in complex OLTP application environments. It does so by exercising a breadth of system components associated with such environments, which are characterized by:

- The simultaneous execution of multiple transaction types that span a breadth of complexity
- On-line and deferred transaction execution modes
- Multiple on-line terminal sessions
- Moderate system and application execution time
- Significant disk input/output
- Transaction integrity (ACID properties)
- Non-uniform distribution of data access through primary and secondary keys
- Databases consisting of many tables with a wide variety of sizes, attributes, and relationships
- Contention of data access and update

The performance metric reported by TPC-C is a “business throughput” measuring the number of orders processed per minute. Multiple transactions are used to simulate the business activity of processing an order, and each transaction is subject to a response time constraint. The performance metric for this benchmark is expressed in transactions-per-minute-C (tpmC). To be compliant with the TPC-C standard, all references to tpmC results must include the tpmC rate, the associated price-per-tpmC, and the availability date of the priced configuration.

Despite the fact that this benchmark offers a rich environment that emulates many OLTP applications, this benchmark does not reflect the entire range of OLTP requirements. In addition, the extent to which a customer can achieve the results reported by a vendor is highly dependent on how closely TPC-C approximates the customer application. The relative performance of systems derived from this benchmark does not necessarily hold for other workloads or environments. Extrapolations to other environments are not recommended.

Benchmark results are highly dependent upon workload, specific application requirements, and systems design and implementation. Relative system performance will vary as a result of these and other factors. Therefore, TPC-C should not be used as a substitute for a specific customer application benchmark when critical capacity planning and/or product evaluation decisions are contemplated.

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark C test conducted on the Compaq Proliant 7000. The operating system used for the benchmark was Window NT 4.0 Enterprise Edition (SP4). The DBMS used was Microsoft SQL Server 7.0 Enterprise Edition.

TPC Benchmark C Metrics

The standard TPC Benchmark C metrics, tpmC (transactions per minute), price per tpmC (five year capital cost per measured tpmC), and the availability date are reported as:

22,478.90 tpmC
\$18.84 per tpmC

The availability date is 1-Mar-99.

Standard and Executive Summary Statements

The following pages contain executive summary of results for this benchmark.

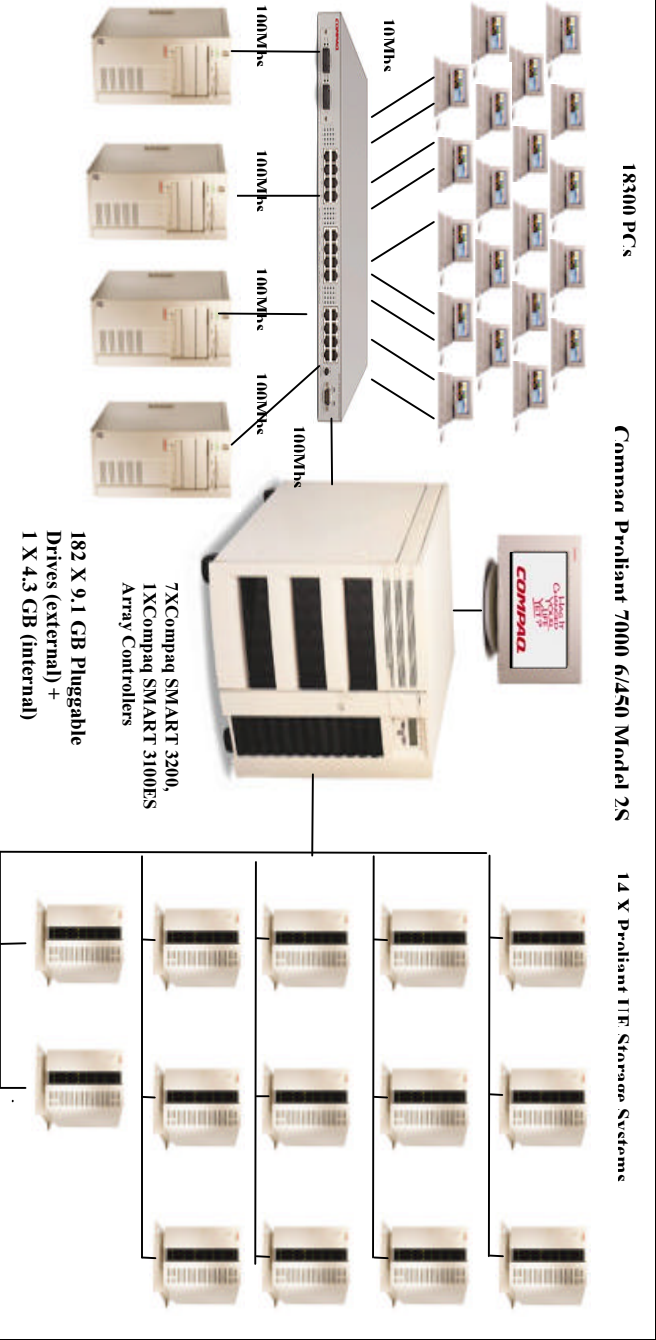
Auditor

The benchmark configuration, environment and methodology were audited by Lorna Livingtree of Performance Metrics, Inc. to verify compliance with the relevant TPC specifications.

Compaq Computer Corporation Proliant 7000-6/450-2S TPC-C Rev. 3.4
 C/S with 4 Proliant 1600 Report Date: Feb 10, 1999

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date
\$423,542	22478.9	\$18.84	Mar 1, 1999

Processors	Database	Operating System	Other Software	Number Users
4 Pentium Xeon 450 MHZ	Microsoft SQL Server 7.0 Enterprise Edition	Windows NT 4.0 Enterprise Edition (SP4)	NonstopTuxedo 1.3 Microsoft Visual C++ MKS Toolkit for NT Microsoft SQL Toolkit	18300



System Components		Quantity	Description	Quantity	Description
Processor		4	450MHz Pentium II Xeon w/ 2MB Cache	2	400MHz Pentium II w/ 512K cache
Memory		8	512MB	2	128MB
Disk Controllers		7	SMART SCSI Array Controller 3200	1	Integrated Ultra Controller
Disk Drives		182	9.1GB SCSI Drive	1	4.3GB SCSI Drive
Total Storage			1458 Gigabytes		
Tape Drives		1	4/16 Turbodat		

Compaq Computer Corporation		Proliant 7000- 6/450-2S		TPC-C REV 3.4		
Client/Server		Report Date:		10-Feb-99		
Description	Part Number	Third Party	Unit Price	Qty	Extended Price	5 Yr. Maint. Price
Server Hardware						
Proliant 7000-6/450-2S 2 x Pentium II Xeon/450MHz - 2MB Cache 3 x Integrated Fast-SCSI-3 Controller CD-ROM, 3100ES SMART Cont. 10/100 Ethernet Controller	386671-001		32,928	1	32,928	
Proliant Warranty Upgrade	276535-000		1,120	1		1,120
Onsite 5x8 4 Hr Response Maintenance						11,962
Proliant Xeon 450 - 2MB Processor Kit	328814-B21		4,816	2	9,632	
512MB EDO DIMM Memory Kit (50ns)	328582-B21		2,005	7	14,034	
Compaq SMART Array Controller-3200	295643-B21		1,960	7	13,720	
Proliant Storage System -UE	348710-002		1,903	14	26,640	
Proliant Storage Warranty Upgrade	276537-001		430	14		6,020
4.3GB UW SCSI-3 10,000 RPM Drive	304858-B21		503	1	503	
9.1 GB Hot-Plug SCSI-3 Drive - 10K	336357-B21		940	182	171,022	
Compaq VGA V55 Color Monitor	308006-001		240	1	240	
4/8GB DAT Drive	295353-B21		703	1	703	
Compaq UPS Model T2000	242688-005		1,786	1	1,786	
Subtotal				1	271,208	19,102
Server Software						
Microsoft SQL Server 7.0 Enterprise		Microsc 2	28,999	1	28,999	10,475
MKS Toolkit for NT		MKS 1	499	1	499	5 Yr. WTY
Microsoft Visual C++ and Subscription Service		Microsc 2	499	1	499	Incl Above
Microsoft Windows NT Enterprise v. 4.0		Microsc 2	3,999	1	3,999	Incl Above
Subtotal				1	33,996	10,475
Client Hardware						
Proliant 1600 6/400 - 64MB, 1001	315570-001		2,496	4	9,986	
Proliant Warranty Upgrade	276535-000		1,120	4		4,480
Onsite 5x8 4 Hr Response Maintenance						2,378
Proliant 1600 6/400 Processor C	313612-B21		1,064	4	4,256	
64-Megabyte SD DIMM Kit	313614-B21		225	4	900	
128-Megabyte SD DIMM Kit	313615-B21		465	8	3,718	
Compaq VGA V55 Color Moni	308006-001		240	4	959	
Subtotal				4	19,820	6,858
Client Software						
Microsoft Windows NT Server v. 4.0		Microsc 2	809	4	3,236	Incl. Above
Microsoft SQL Toolkit		Microsc 2	499	1	499	Incl. Above
Compaq Non-Stop Tuxedo OS for NT - 1.3			1,950	4	7,800	640
Subtotal				4	11,535	640
User Connectivity						
Compaq SW3322 24Port Dual Spd Switch	356300-B21		2,419	4	9,677	See Note 4
8-port 10BaseT hub		Generic 1	29.00	2560	73,950	See Note 4
Subtotal					83,627	0
Large Purchase and Cash discount	9.0%					
					(\$33,719)	
Total					\$386,466	\$37,075
Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark pricing specifications. If you find that the stated prices are not available according to these terms, please inform the TPC at pricing@tpc.org. Thank you.						
(Pricing: 1=Alisat Systems 2=Microsoft) - Note 4 = 5 Year warranty with 10% Spares						
Note: The benchmark results and test methodology were audited by Loma Livingtree of Performance Metrics, Inc.						
Five-Year Cost of Ownership: \$423,542						
tpmC Rating: 22478.90						
\$ / tpmC: \$18.84						

Numerical Quantities Summary

MQTH, Computed Maximum Qualified Throughput

22478.9 tpmC

% throughput difference, reported & reproducibility runs

0.10%

Response Times (in seconds)	Average	90%	Maximum
New-Order	0.63	1.11	6.74
Payment	0.51	0.99	7.45
Order-Status	0.54	1.01	6.32
Delivery (interactive portion)	0.11	0.11	5.74
Delivery (deferred portion)	0.47	0.77	2.38
Stock-Level	2.08	2.82	8.52
Menu	0.11	0.12	6.11

Transaction Mix, in percent of total transaction

New-Order	44.76%
Payment	43.13%
Order-Status	4.04%
Delivery	4.02%
Stock-Level	4.05%

Emulation Delay (in seconds)

	Resp. Time	Menu
New-Order	N/A	N/A
Payment	N/A	N/A
Order-Status	N/A	N/A
Delivery (interactive)	N/A	N/A
Stock-Level	N/A	N/A

Keying/Think Times (in seconds)

	Min.	Average	Max.
New-Order	18.00/0.00	18.00/12.17	18.22/50.90
Payment	3.00/0.00	3.00/12.17	3.10/123.28
Order-Status	2.00/0.00	2.01/12.16	2.06/123.27
Delivery (interactive)	2.01/0.00	2.01/5.12	2.08/50.90
Stock-Level	2.01/0.00	2.01/10.08	2.08/101.90

Test Duration

Ramp-up time	60 minutes
Measurement interval	20 minutes
Transactions (all types) completed during measurement interval	997336
Ramp down time	11 minutes

Checkpointing

Number of checkpoints	1
Checkpoint interval	20 minutes

General Items

Test Sponsor

A statement identifying the benchmark sponsor(s) and other participating companies must be provided.

This benchmark was sponsored by Compaq Computer Corporation. The benchmark was developed and engineered by Compaq Computer Corporation. Testing took place at Compaq benchmarking laboratories in Houston, Texas.

Application Code and Definition Statements

The application program (as defined in clause 2.1.7) must be disclosed. This includes, but is not limited to, the code implementing the five transactions and the terminal input output functions.

Appendix A contains all source code implemented in this benchmark.

Parameter Settings

Settings must be provided for all customer-tunable parameters and options which have been changed from the defaults found in actual products, including by not limited to:

- Database options
- Recover/commit options
- Consistency locking options
- Operating system and application configuration parameters

This requirement can be satisfied by providing a full list of all parameters.

Appendix C contains the tunable parameters to for the database, the operating system, and the transaction monitor.

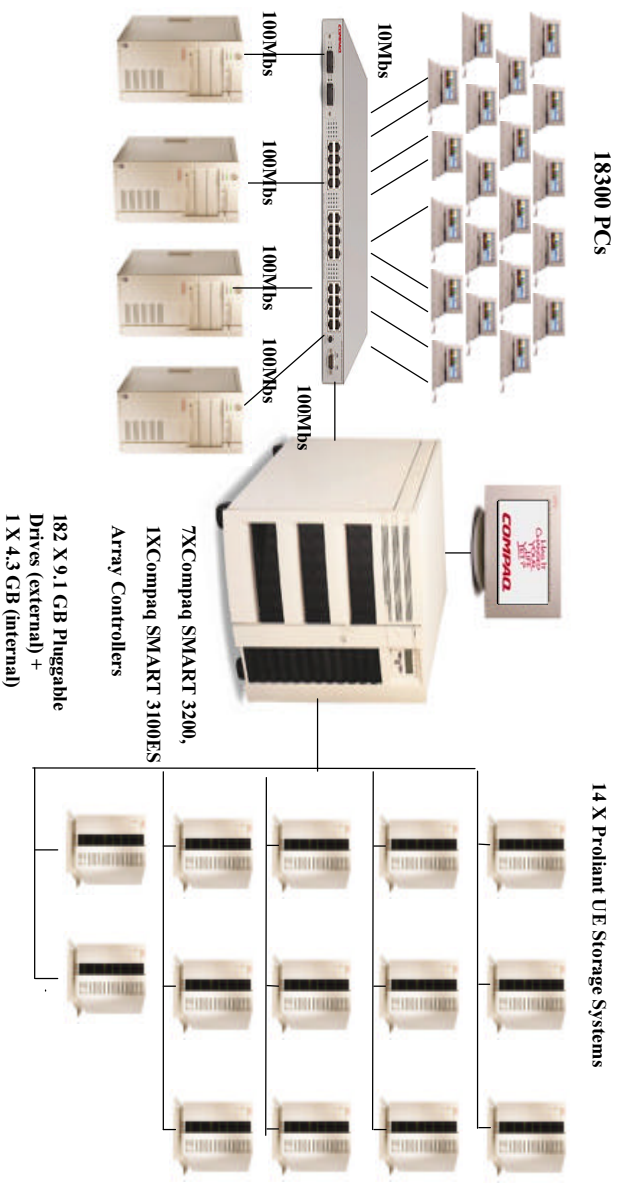
Configuration Items

Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences.

The configuration diagrams for both the tested and priced systems are included on the following pages.

Figure 2. Priced Configuration

Compaq Proliant 7000 6/450 Model 2S



Clause 1 Related Items

Table Definitions

Listing must be provided for all table definition statements and all other statements used to set up the database.

Appendix B contains the code used to define and load the database tables.

Physical Organization of Database

The physical organization of tables and indices within the database must be disclosed.

The tested configuration consisted of: 182 drives at 9.1GB each and 1 drive at 4.3GB.

Benchmarked Configuration:

Embedded Controller 1,

EISA UTILITIES PARTITION

Compaq System Configuration Utilities

LOGICAL DRIVE C:

Microsoft Windows NT v. 4.0 SP4

Total Capacity = 39 MB

Total Capacity = 4055 MB

SMART-3200 Controller, Slot 3, Array A

LOGICAL DRIVE G:

Big_fg

LOGICAL DRIVE N:

Misc_fg

LOGICAL DRIVE Y:

Tpback1

Total Capacity = 17033 MB

Total Capacity = 12997 MB

Total Capacity = 170810 MB

RAID 0

RAID 0

RAID 5

SMART-3200 Controller, Slot 4, Array A

LOGICAL DRIVE H:

Big_fg

LOGICAL DRIVE O:

Misc_fg

LOGICAL DRIVE Z:

Tpback2

Total Capacity = 17511 MB

Total Capacity = 12997 MB

Total Capacity = 170347 MB

RAID 0

RAID 0

RAID 5

SMART-3200 Controller, Slot 5, Array A

LOGICAL DRIVE M:

Big_fg

LOGICAL DRIVE T:

Misc_fg

Total Capacity = 21005 MB

Total Capacity = 12997 MB

RAID 0

RAID 0

SMART-3200 Controller, Slot 6, Array A

LOGICAL DRIVE I:

Big_fg

LOGICAL DRIVE P:

Misc_fg

Total Capacity = 20403 MB

Total Capacity = 12997 MB

RAID 0

RAID 0

SMART-3200 Controller, Slot 7, Array A

<u>LOGICAL DRIVE J:</u>	<u>Total Capacity = 21005 MB</u>	<u>RAID 0</u>
<u>Big_fg</u>		
<u>LOGICAL DRIVE Q:</u>	<u>Total Capacity = 12997 MB</u>	<u>RAID0</u>
<u>Misc_fg</u>		

SMART-3200 Controller, Slot 8, Array A	<u>Total Capacity = 20702 MB</u>	<u>RAID 0</u>
<u>LOGICAL DRIVE K:</u>		
<u>Big_fg</u>		
<u>LOGICAL DRIVE R:</u>	<u>Total Capacity = 12997 MB</u>	<u>RAID0</u>
<u>Misc_fg</u>		

SMART-3200 Controller, Slot 9, Array A	<u>Total Capacity = 20802 MB</u>	<u>RAID 0</u>
<u>LOGICAL DRIVE L:</u>		
<u>Big_fg</u>		
<u>LOGICAL DRIVE S:</u>	<u>Total Capacity = 12997 MB</u>	<u>RAID0</u>
<u>Misc_fg</u>		

SMART-3100ES Controller, Slot 11, Array A	<u>Total Capacity = 52063 MB</u>	<u>RAID 1</u>
<u>LOGICAL DRIVE F:</u>		
log		

Priced Configuration vs. Measured Configuration:

The priced configuration and the tested configuration differ in that the benchmarked configuration was run on a Proliant 7000Rack Mountable System with Rack Mountable Proliant Storage Systems, whereas the priced configuration includes all tower (non-rack) Proliant Storage Systems and a tower Proliant 7000server.

Insert and Delete Operations

It must be ascertained that insert and/or delete operations to any of the tables can occur concurrently with the TPC-C transaction mix. Furthermore, any restrictions in the SUT database implementation that precludes inserts beyond the limits defined in Clause 1.4.11 must be disclosed. This includes the maximum number of rows that can be inserted and the minimum key value for these new rows.

All insert and delete functions were fully operational during the entire benchmark.

Partitioning

While there are a few restrictions placed upon horizontal or vertical partitioning of tables and rows in the TPC-C benchmark, any such partitioning must be disclosed.

No partitioning was used in this benchmark.

Replication, Duplication or Additions

Replication of tables, if used, must be disclosed. Additional and/or duplicated attributes in any table must be disclosed along with a statement on the impact on performance.

No replications, duplications or additional attributes were used in this benchmark.

Clause 2 Related Items

Random Number Generation

The method of verification for the random number generation must be described.

The driver used the Performix RTE from Pure Atria, Inc. to generate random numbers for the input data. The random number generator received a unique seed for each user.

The Performix RTE computes random integers using an implementation of a linear congruential sequence as described in “The Art of Computer Programming, Volume 2/ Seminumerical Algorithms” by Donald E. Knuth, copyright 1981 by Addison-Wesley Publishing Company.

The seeds for each user were captured and verified by the auditor to be unique. In addition, the contents of the database were systematically searched, and randomly sampled by the auditor for patterns that would indicate the random number generator had effected any kind of a discernible pattern; none were found.

Input/Output Screen Layout

The actual layout of the terminal input/output screens must be disclosed.

All screen layouts followed the specifications exactly.

Priced Terminal Feature Verification

The method used to verify that the emulated terminals provide all the features described in Clause 2.2.2.4 must be explained. Although not specifically priced, the type and model of the terminals used for the demonstration in 8.1.3.3 must be disclosed and commercially available (including supporting software and maintenance).

The terminal attributes were verified by the auditor manually exercising each specification on a representative Proliant 1600 6/400 server.

Presentation Manager or Intelligent Terminal

Any usage of presentation managers or intelligent terminals must be explained.

Application code running on the client machines implemented the TPC-C user interface. No presentation manager software or intelligent terminal features were used. The source code for the forms applications is listed in Appendix A.

Transaction Statistics

Table 2.1 lists the numerical quantities that Clauses 8.1.3.5 to 8.1.3.11 require.

Table 2.1 Transaction Statistics

	Statistic	Value
New Order	Home warehouse order lines	90.03%
	Remote warehouse order lines	9.97%
	Rolled back transactions	1.00%
	Average items per order	9.99%
Payment	Home warehouse payments	85.06%
	Remote warehouse payments	14.94%
	Accessed by last name	59.99%
Order Status	Accessed by last name	59.86%
Transaction Mix	New Order	44.76%
	Payment	43.13%
	Order status	4.04%
	Delivery	4.02%
	Stock level	4.05%

Queuing Mechanism

The queuing mechanism used to defer the execution of the Delivery transaction must be disclosed.

NonStop Tuxedo 1.3 on each client machine served as the queuing mechanism to the database. Each delivery request was submitted to Tuxedo asynchronously with control being returned to the client process immediately and the deferred delivery part completing asynchronously.

The source code is listed in Appendix A.

Clause 3 Related Items

Transaction System Properties (ACID)

The results of the ACID tests must be disclosed along with a description of how the ACID requirements were met. This includes disclosing which case was followed for the execution of Isolation Test 7.

All ACID property tests were successful. The executions are described below.

Atomicity

The system under test must guarantee that the database transactions are atomic; the system will either perform all individual operations on the data or will assure that no partially completed operations leave any effects on the data.

Completed Transactions

A row was selected in a script from the warehouse, district and customer tables, and the balances noted. A payment transaction was started with the same warehouse, district and customer identifiers and a known amount. The payment transaction was committed and the rows were verified to contain correctly updated balances.

Aborted Transactions

A row was selected in a script from the warehouse, district and customer tables, and the balances noted. A payment transaction was started with the same warehouse, district and customer identifiers and a known amount. The payment transaction was rolled back and the rows were verified to contain the original balances.

Consistency

Consistency is the property of the application that requires any execution of a database transaction to take the database from one consistent state to another, assuming that the database is initially in a consistent state.

Consistency conditions one through four were tested using a shell script to issue queries to the database. The results of the queries verified that the database was consistent for all four tests.

A run was executed under full load lasting over two hours and included a checkpoint.

The shell script was executed again. The result of the same queries verified that the database remained consistent after the run.

Isolation

Sufficient conditions must be enabled at either the system or application level to ensure the required isolation defined above (clause 3.4.1) is obtained.

Isolation tests one through seven were executed using shell scripts to issue queries to the database. Each script included timestamps to demonstrate the concurrency of operations. The results of the queries were captured to files. The captured files were verified by the auditor to demonstrate the required isolation had been met.

In addition, the phantom tests and the stock level tests were executed and verified.

For Isolation test seven, case A was followed.

Durability

The tested system must guarantee durability: the ability to preserve the effects of committed transaction and insure database consistency after recovery from any one of the failures listed in Clause 3.5.3.

Durable Media Failure

Durability from media failure was demonstrated on a database scaled for 10 warehouses. The standard driving mechanism was used to generate the transaction load of 100 users. The fully scaled database under full load would also have passed the following test. Loss of data and loss of log were combined in the same test.

Loss of Data and Log

To demonstrate recovery from a permanent failure of durable medium containing DBMS logs and TPC-C tables, the following steps were executed:

- The database was backed up to extra disks.
- The total number of New Orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving the beginning count.
- The RTE was started with 100 users.
- The test was allowed to run for a minimum of 10 minutes.
- One log disk was removed from the drive cabinet.
- Since the disk was mirrored, processing was not interrupted. This was verified by checking the users status on the RTE.
- One of the data disks was removed from the drive cabinet.
- When Microsoft SQL Server recorded errors about not being able to access the database, the RTE was shut down.
- A dump of the transaction log was taken and the Microsoft SQL Server was shutdown.
- A new log disk was inserted into the log drive cabinet. A new data disk was inserted into the data drive cabinet. Both cabinets were powered back up and system restarted.
- Microsoft SQL Server was started.
- The database was automatically recovered from the database backup and the transaction log dump.
- Consistency condition #3 was executed and verified.
- Step 2 was repeated and the difference between the first and second counts was noted.
- An RTE report was generated for the entire run time giving the number of NEW-ORDERS successfully returned to the RTE.
- The counts in step 12 and 13 were compared and the results verified that all committed transactions had been successfully recovered.
- Samples were taken from the RTE files and used to query the database to demonstrate successful transactions had corresponding rows in the ORDER table.

Instantaneous Interruption and Loss of Memory

Because loss of power erases the contents of memory, the instantaneous interruption and the loss of memory tests were combined into a single test. This test was executed on a fully scaled database of 1830 warehouses under a full load of 18300 users. The following steps were executed:

- The total number of New Orders was determined by the sum of D_NEXT_O_ID of all rows in the DISTRICT table giving the beginning count.
- The RTE was started with 18300 users.
- The test was allowed to run for a minimum of 10 minutes.
- A checkpoint was performed.
- System crash and loss of memory were induced by switching the power off. No battery backup or Uninterruptible Power Supply (UPS) were used to preserve the contents of memory.
- The RTE was shutdown.

- Power was restored and the system restarted.
- Microsoft SQL Server was restarted and performed an automatic recovery.
- Consistency condition #3 was executed and verified.
- Step 1 was repeated and the difference between the first and second counts was noted.
- An RTE report was generated for the entire run time giving the number of NEW-ORDERS successfully returned to the RTE.
- The counts in step 9 and 10 were compared and the results verified that all committed transactions had been successfully recovered.

Samples were taken from the RTE files and used to query the database to demonstrate successful transactions had corresponding rows in the ORDER table.

Clause 4 Related Items

Initial Cardinality of Tables

The cardinality (e.g. number of rows) of each table, as it existed at the start of the benchmark run, must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted, the cardinality of the WAREHOUSE table as initially configured and the number of rows deleted must be disclosed.

Table 4.1 Number of Rows for Server

Table	Cardinality as built
Warehouse	1830
District	18300
Customer	54900000
History	54900000
Orders	54900000
NewOrder	16470000
OrderLine	549006634
Stock	183000000
Item	100000
Deleted Warehouses	0

Database Layout

The distribution of tables and logs across all media must be explicitly depicted for tested and priced systems.

The benchmarked configuration used 7 SMART-3200 Array controllers, each with 2 SCSI channels and one SMART-3100ES Array controller with 3 SCSI channels. Each SMART-3200 controller is capable of accessing up to 24 disk drives, 12 disk drives per each channel, and supports RAID 0, RAID 1 and RAID 5 per each logical volume configured. The data tables were stored on 14 logical volumes of 24 drives each (i.e. 2 logical volumes on each controller of 24 drives). Each data volume was configured with RAID 0 and the Array Accelerator was enabled on the second logical volume of each controller (100% write cache). One logical volume of twelve 9.1GB drives was configured with RAID 1 and stored the transaction log with 2 additional 9.1GB drives on the controller but not configured. The transaction log volumes had the Array Accelerator disabled. The operating system was stored on a 4.3GB drive on an embedded channel. All RAID volumes used hardware RAID.

Section 1.2 of this report details the distribution of database tables across all disks. The code that creates the filegroups and tables is included in Appendix B.

Type of Database

A statement must be provided that describes:

- *The data model implemented by DBMS used (e.g. relational, network, hierarchical).*
- *The database interface (e.g. embedded, call level) and access language (e.g. SQL, DL/I, COBOL read/write used to implement the TPC-C transaction. If more than one interface/access language is used to implement TPC-C, each interface/access language must be described and a list of which interface/access language is used with which transaction type must be disclosed.*

Microsoft SQL Server 7.0 Enterprise Edition is a relational DBMS.

The interface used was MS SQL Server stored procedures accessed with Remote Procedure Calls embedded in C code.

Database Mapping

The mapping of database partitions/replications must be explicitly described.

The database was not replicated.

180 Day Space

Details of the 180 day space computations along with proof that the database is configured to sustain 8 hours of growth for the dynamic tables (Order, Order-Line, and History) must be disclosed.

To calculate the space required to sustain the database log for 8 hours of growth at steady state, the following steps were followed:

- The free space on the log file was queried using *DBCC checktable(syslogs)*.
- Transactions were run against the database with a full load of users.
- The free space was again queried using *DBCC checktable(syslogs)*.
- The space used was calculated as the difference between the first and second query.
- The number of NEW-ORDERS was verified from an RTE report covering the entire run.
- The space used was divided by the number of NEW-ORDERS giving a space used per NEW-ORDER transaction.
- The space used per transaction was multiplied by the measured tpmC rate times 480 minutes.

The same methodology was used to compute growth requirements for dynamic tables Order, Order-Line and History.

The details of both the 8-hour transaction log space requirement and the 180-day space requirement is shown in Appendix D.

Clause 5 Related Items

Throughput

Measured tpmC must be reported

Measured tpmC 22,478.90 tpmC
Price per tpmC \$18.84 per tpmC

Response Times

Ninetieth percentile, maximum and average response times must be reported for all transaction types as well as for the menu response time.

Table 5.2: Response Times

Type	Average	Minimum	90th%
NewOrder	063	674	1.11
Payment	051	745	099
OrderStatus	054	632	101
Interactive Delivery	011	574	011
Deferred Delivery	047	238	077
StockLevel	208	852	282
Menu	011	611	012

Keying and Think Times

The minimum, the average, and the maximum keying and think times must be reported for each transaction type.

Table 5.3: Keying Times

Type	Minimum	Average	Maximum
NewOrder	1800	1800	1822
Payment	300	300	310
OrderStatus	200	201	206
Interactive Delivery	201	201	208
StockLevel	201	201	208

Table 5.4: Think Times

Type	Minimum	Average	Maximum
New-Order	0.00	12.17	123.28
Payment	0.00	12.16	123.27
Order-Status	0.00	10.08	101.90
Interactive Delivery	0.00	5.11	50.90
Stock-Level	0.00	5.15	48.10

Response Time Frequency Distribution Curves and Other Graphs

Response Time frequency distribution curves (see Clause 5.6.1) must be reported for each transaction type.

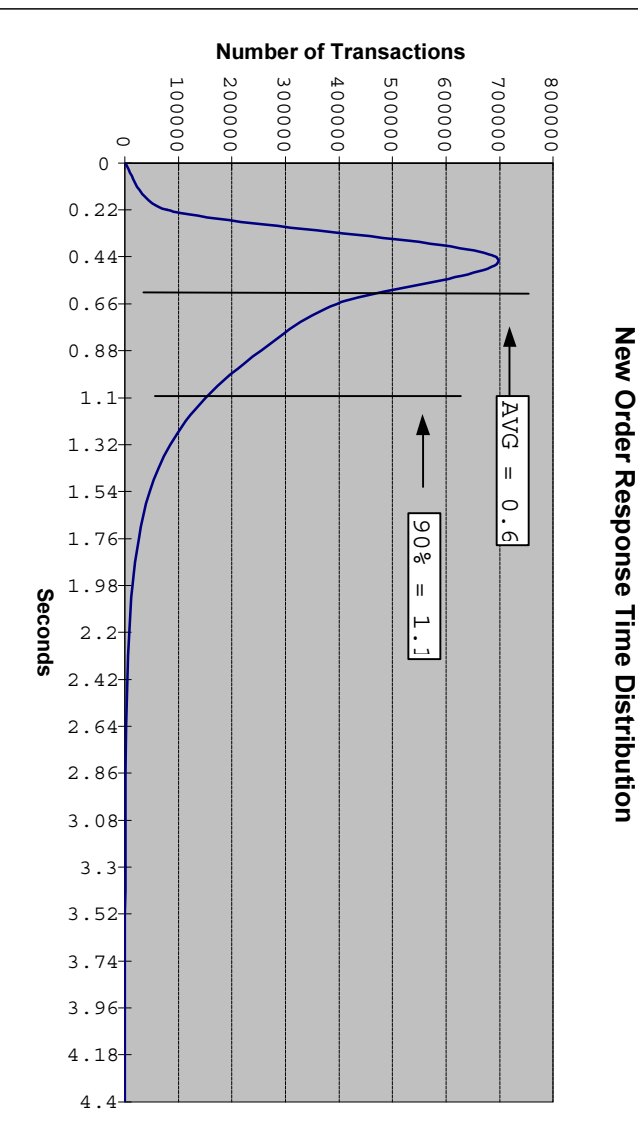
The performance curve for response times versus throughput (see Clause 5.6.2) must be reported for the New-Order transaction.

Think Time frequency distribution curves (see Clause 5.6.3) must be reported for each transaction type.

Keying Time frequency distribution curves (see Clause 5.6.4) must be reported for each transaction type.

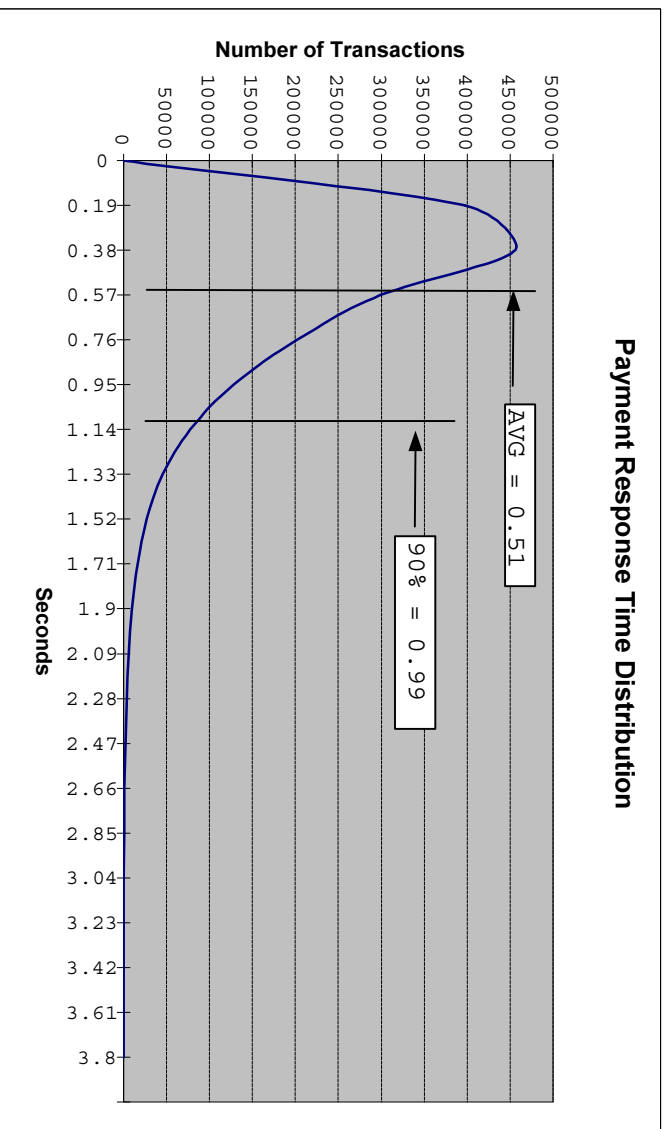
A graph of throughput versus elapsed time (see Clause 5.6.5) must be reported for the New-Order transaction.

Figure 3. New Order Response Time Distribution



New Order Response Time Distribution

Figure 4. Payment Response Time Distribution



Payment Response Time Distribution

Figure 5. Order Status Response Time Distribution

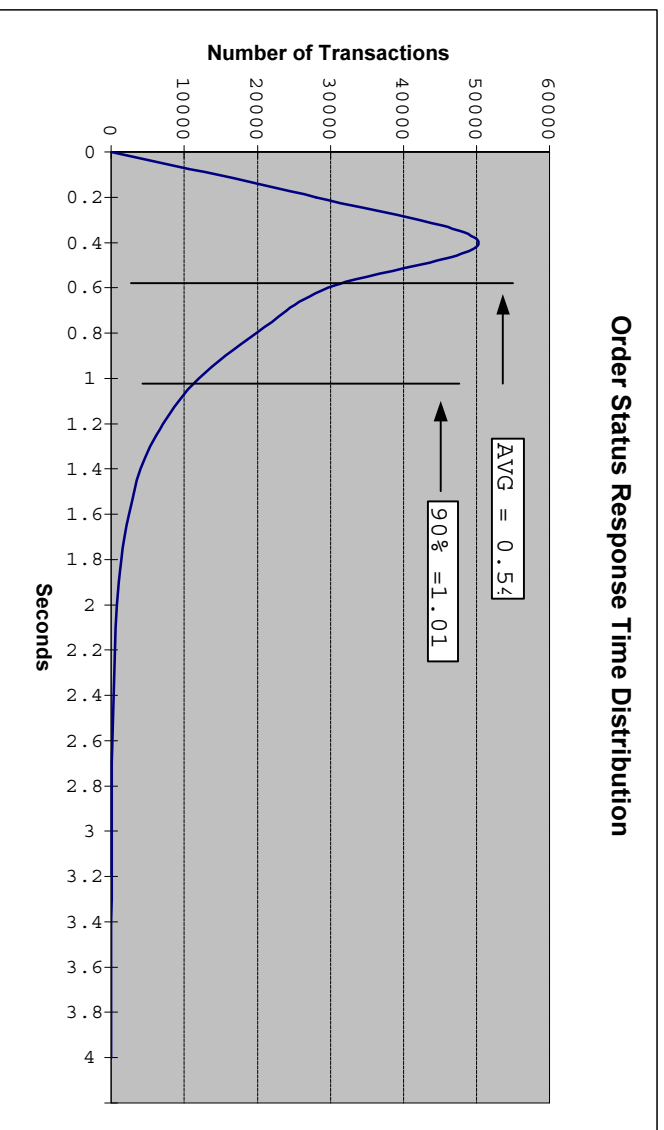


Figure 6. Delivery Response Time Distribution

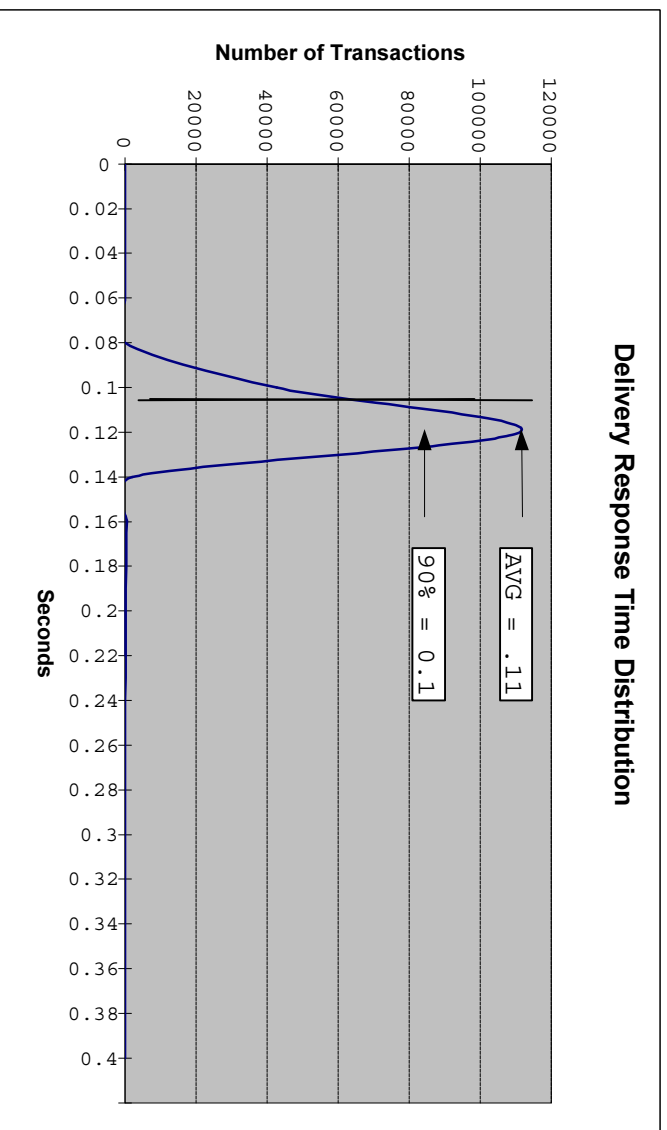


Figure 7. Stock Level Response Time Distribution

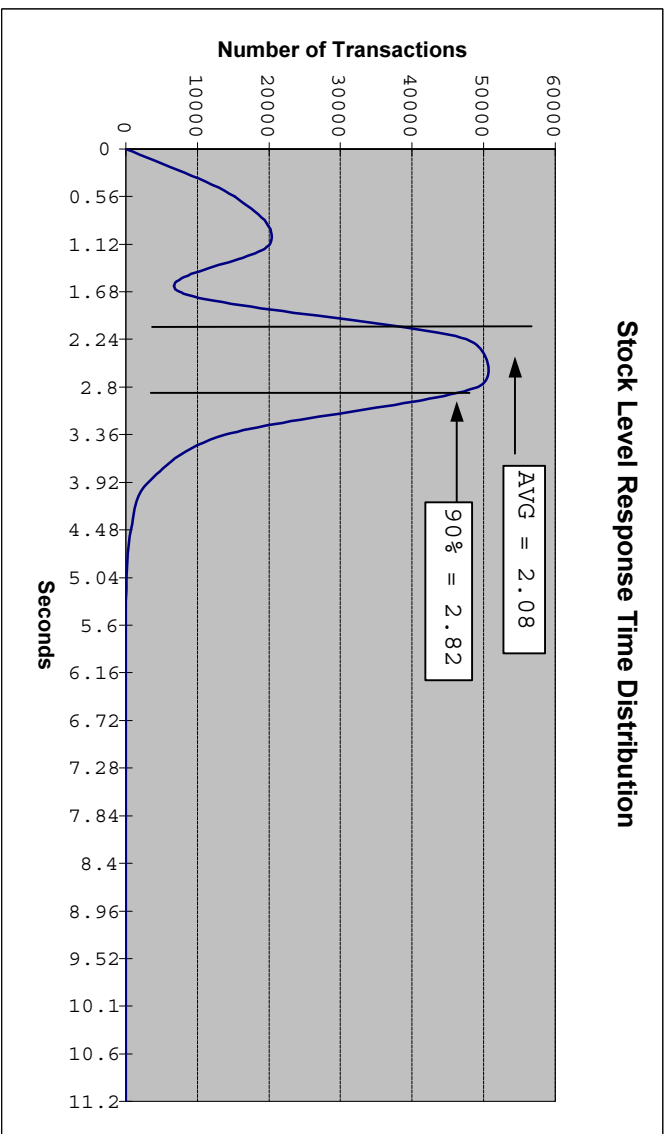
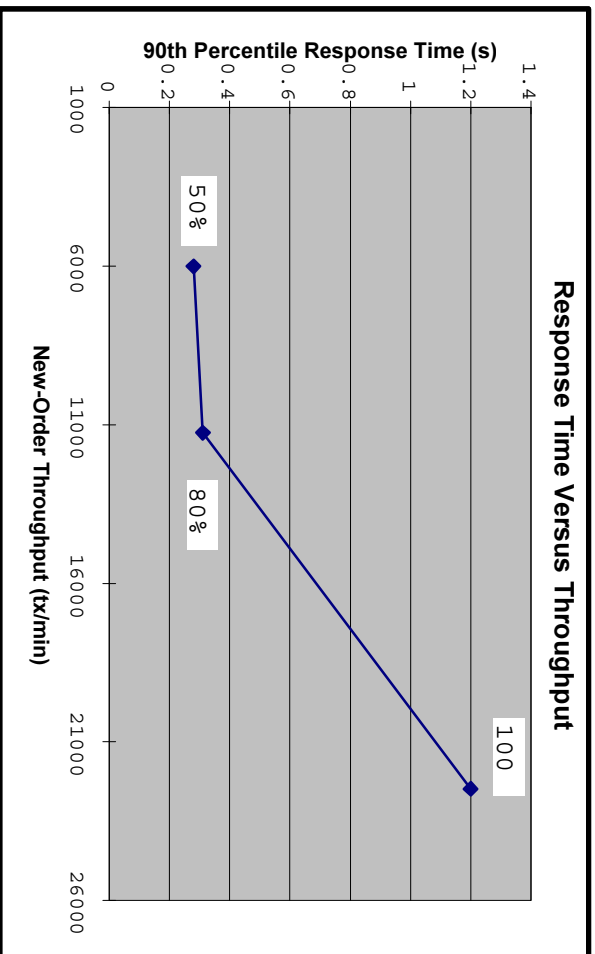


Figure 8. Response Time vs. Throughput



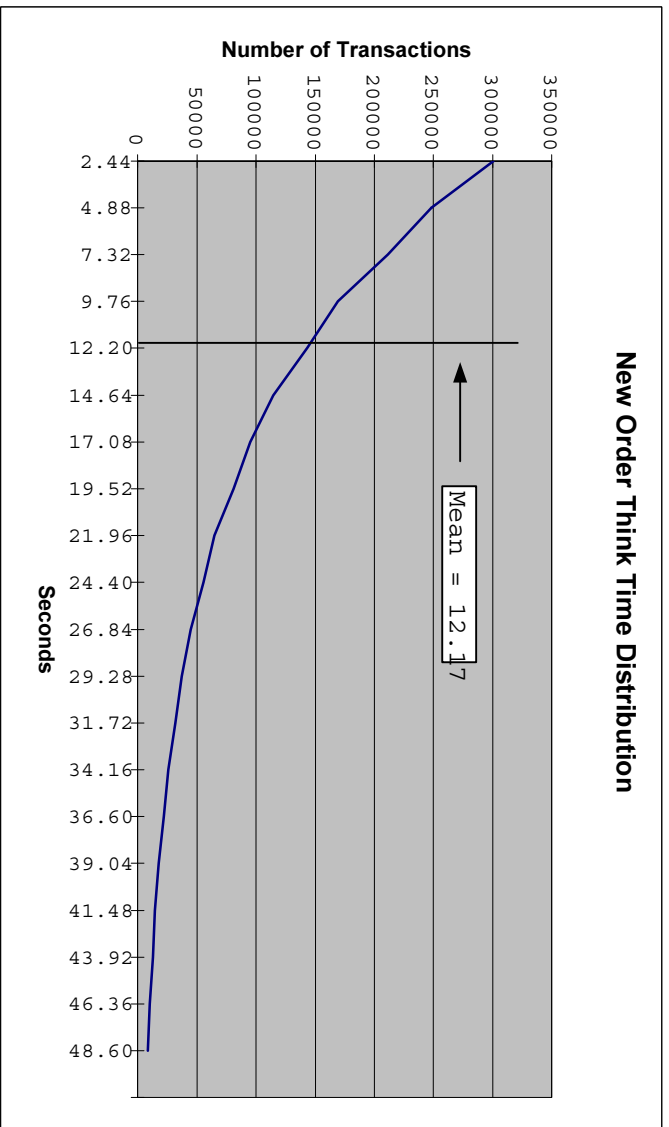


Figure 9. New Order Think Time Distribution

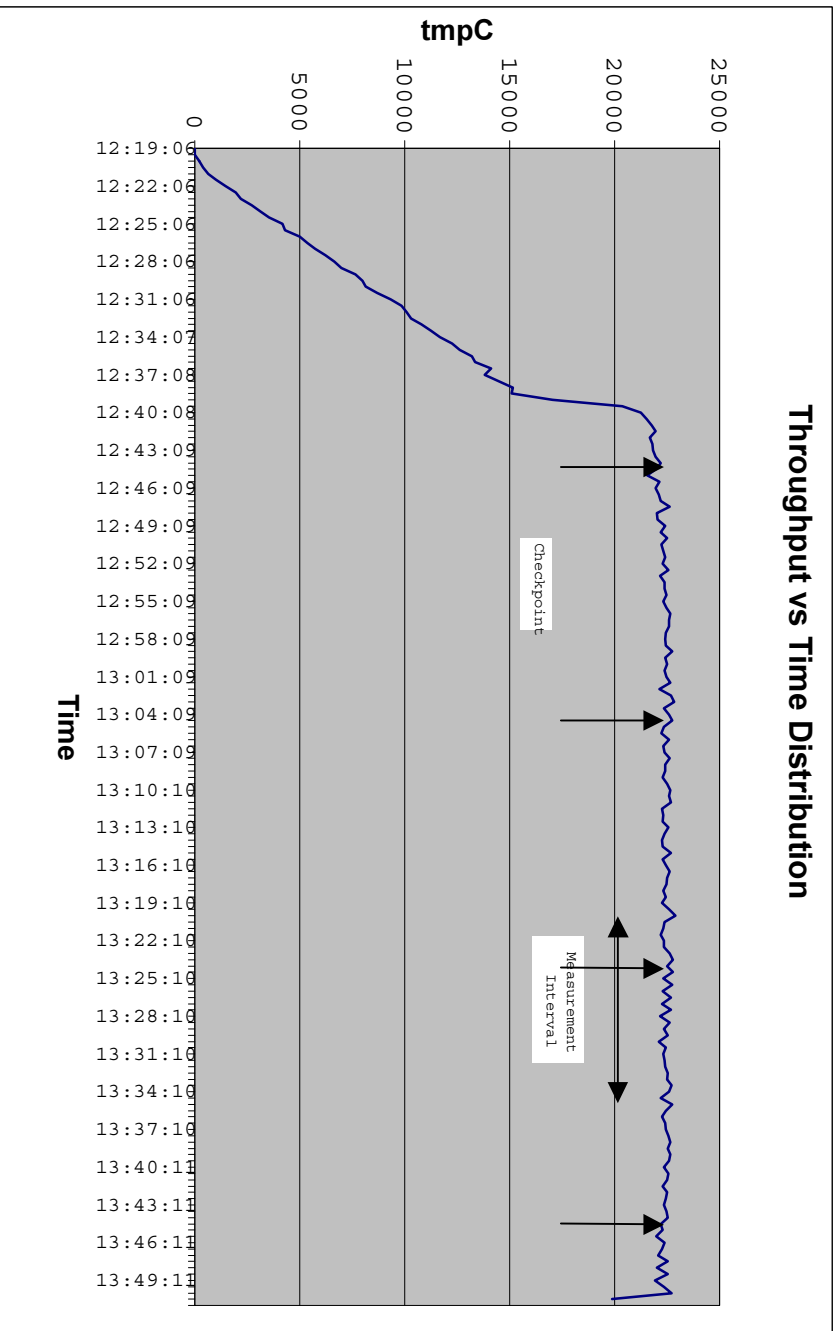


Figure 10. Throughput vs. Time Distribution

Steady State Determination

The method used to determine that the SUT had reached a steady state prior to commencing the measurement interval must be disclosed.

Steady state was determined using real time monitor utilities from the RTE. Steady state was further confirmed by the throughput data collected during the run and graphed in Figure 5.8.

Work Performed During Steady State

A description of how the work normally performed during a sustained test (for example checkpointing, writing redo/undo log records, etc.), actually occurred during the measurement interval must be reported.

The RTE generated the required input data to choose a transaction from the menu. This data was timestamped and captured in RTE log files before being transmitted. There was one log file for each user. The input screen for the requested transaction was returned and it was also captured and timestamped in the RTE log files. The difference between these two timestamps was the menu response time.

The RTE generated the required input data for the chosen transaction. It waited to complete the minimum required key time before transmitting the input screen. The transmission was timestamped and captured in RTE log files. The return of the screen with the required response data was timestamped and captured in the RTE log files. The difference between these two timestamps was the response time for that transaction.

The RTE then waited the required think time interval before repeating the process starting at selecting a transaction from the menu.

The RTE transmissions were sent to application processes running on the client machines through Ethernet LANs. These client application processes handled all screen I/O as well as all requests to the database on the server. The applications communicated with the database server over another Ethernet LAN using DBLIB and RPC calls.

To perform checkpoints at specific intervals, we set SQL Server *recovery interval* to the maximum allowable value and wrote a script to schedule multiple checkpoints at specific intervals. The script logged the checkpoint beginning and ending time in the chkpt.out output file. The script included a wait time between each checkpoint equal to the measurement interval which was 30 minutes. The checkpoint script was started manually after the RTE had all users logged in and sending transactions.

At each checkpoint, MS SQL Server wrote to disk all memory pages that had been updated but not yet physically written to disk. The positioning of the measurement interval was verified to be clear of the guard zones and is depicted on the graph in Figure 5.8.

Reproducibility

A description of the method used to determine the reproducibility of the measurement results must be reported.

We allowed the database to warm up and to reach a steady state for approximately 60 minutes. The steady state was sustained for a 20-minute (measurement) interval, and was followed by a ramp-down. The repeatable interval result was within 0.1 % of the reported interval result.

Measurement Period Duration

A statement of the duration of the measurement interval for the reported Maximum Qualified Throughput (mpmC) must be included.

The reported measured interval was exactly 20 minutes long.

Regulation of Transaction Mix

The method of regulation of the transaction mix (e.g., card decks or weighted random distribution) must be described. If weighted distribution is used and the RTE adjusts the weights associated with each transaction type, the maximum adjustments to the weight from the initial value must be disclosed.

The RTE was given a weighted random distribution, which could not be adjusted during the run.

Transaction Statistics

The percentage of the total mix for each transaction type must be disclosed. The percentage of New-Order transactions rolled back as a result of invalid item number must be disclosed. The average number of order-lines entered per New-Order transaction must be disclosed. The percentage of remote order lines per New-Order transaction must be disclosed. The percentage of remote Payment transactions must be disclosed. The percentage of customer selections by customer last name in the Payment and Order-Status transactions must be disclosed. The percentage of Delivery transactions skipped due to there being fewer than necessary orders in the New-Order table must be disclosed.

Table 5.5: Transaction Statistics

	Statistic	Value
New Order	Home warehouse order lines	90.03%
	Remote warehouse order lines	9.97%
	Rolled back transactions	1.00%
	Average items per order	9.99%
Payment	Home warehouse payments	85.06%
	Remote warehouse payments	14.94%
	Accessed by last name	59.99%
Order Status	Accessed by last name	59.86%
Transaction Mix	New Order	44.76%
	Payment	43.13%
	Order status	4.04%
	Delivery	4.02%
	Stock Level	4.05%

Checkpoint Count and Location

The number of checkpoints in the Measurement Interval, the time in seconds from the start of the Measurement Interval to the first checkpoint, and the Checkpoint Interval must be disclosed.

The initial checkpoint was started 25 minutes after the start of the ramp-up. Subsequent checkpoints occurred every 20 minutes. The checkpoint in the measurement interval lasted approximately 5 minutes. The measurement interval contains the third checkpoint, and is clear of the guard zones.

Clause 6 Related Items

RTE Descriptions

If the RTE is commercially available, then its inputs must be specified. Otherwise, a description must be supplied of what inputs (e.g., scripts) to the RTE had been used.

The RTE used was Performix by Pure Atria, Inc. and the code used with Performix is included in Appendix A.

Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to the priced system. The results of the test described in Clause 6.6.3.4 must be disclosed.

The driver system consisted of 20 850R servers. These driver machines were attached to the 4 Compaq Proliant 1600 6/400 client machines through a 10 Mb/s Ethernet connection to the Compaq SW3322 Dual Speed Switch. Since this configuration is exactly the same connectivity as the priced system, no components were being emulated. Therefore, the test described in Clause 6.6.3.4 was not required.

Functional Diagrams

A complete functional diagram of both the benchmark configuration and the configuration of the proposed (target) system must be disclosed. A detailed list of all hardware and software functionality being performed on the Driver System and its interface to the SUT must be disclosed.

The driver system performed the data generation and input functions of the priced display device. It also captured the input and output data and timestamps for post-processing of the reported metrics. No other functionality was included on the driver system.

Section 1.4 of this report contains detailed diagrams of both the benchmark configuration and the priced configuration.

Networks

The network configuration of both the tested services and proposed (target) services which are being represented and a thorough explanation of exactly which parts of the proposed configuration are being replaced with the Driver System must be disclosed.

The bandwidth of the networks used in the tested/priced configuration must be disclosed.

In the tested configuration, the 20 driver (RTE) machines were evenly distributed across two Compaq SW3322 24-port Dual Speed switches and were connected to them with 10 Mbps Ethernet connections. The four client machines were also evenly distributed across the same two Compaq switches and were connected to them with 100 Mbps Ethernet connections. The server (SUT) was connected to one of the Compaq switches with a 100 Mbps Ethernet connection. A crossover cable was run between the two switches to allow ethernet traffic to pass between the switches.

Operator Intervention

If the configuration requires operator intervention (see Clause 6.6.6), the mechanism and the frequency of this intervention must be disclosed.

This configuration does not require any operator intervention to sustain eight hours of the reported throughput.

Clause 7 Related Items

System Pricing

A detailed list of hardware and software used in the priced system must be reported. Each separately orderable item must have vendor part number, description, and release/revision level, and either general availability status or committed delivery data. If package-pricing is used, vendor part number of the package and a description uniquely identifying each of the components of the package must be disclosed. Pricing source and effective date(s) of price(s) must also be reported.

The total 5 year price of the entire configuration must be reported, including: hardware, software, and maintenance charges. Separate component pricing is recommended. The basis of all discounts used must be disclosed.

The details of the hardware and software are reported in the front of this report as part of the executive summary. All third party quotations are included at the end of this report as Appendix E.

Availability, Throughput, and Price Performance

The committed delivery date for general availability (availability date) of products used in the price calculation must be reported. When the priced system included products with different availability dates, the reported availability date for the priced system must be the date at which all components are committed to be available.

A statement of the measured tpmC as well as the respective calculations for the 5-year pricing, price/performance (price/tpmC), and the availability date must be included.

- **Maximum Qualified Throughput** 22,478.90 tpmC
- **Price per tpmC** \$18.84 per tpmC
- **Availability** 1-Mar-1999

Country Specific Pricing

Additional Clause 7 related items may be included in the Full Disclosure Report for each country specific priced configuration. Country specific pricing is subject to Clause 7.1.7

This system is being priced for the United States of America.

Usage Pricing

For any usage pricing, the sponsor must disclose:

- Usage level at which the component was priced.
- A statement of the company policy allowing such pricing.

The component pricing based on usage is shown below:

- 5 Microsoft Windows NT 4.0 SP4 licenses
- One Microsoft SQL Server 7.0 Enterprise Edition (Unlimited User Licenses)
- One Microsoft SQL Workstation (includes SQL Server Programmers Toolkit)
- One Microsoft Visual C++
- Compaq Servers include 3 years of support.

Clause 9 Related Items

Auditor's Report

The auditor's name, address, phone number, and a copy of the auditor's attestation letter indicating compliance must be included in the Full Disclosure Report.

This implementation of the TPC Benchmark C was audited by Lorra Livingtree of Performance Metrics, Inc.

Performance Metrics, Inc.
2229 Benita Drive, Suite 101
Rancho Cordova, CA 95670
(phone) (916) 635-2822
(fax) (916) 858-0109
e-mail: lorna@perfmetrics.com

Availability of the Full Disclosure Report

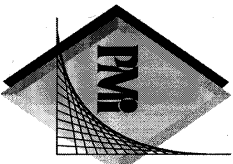
The Full Disclosure Report must be readily available to the public at a reasonable charge, similar to the charges for similar documents by the test sponsor. The report must be made available when results are made public. In order to use the phrase "TPC Benchmark™ C", the Full Disclosure Report must have been submitted to the TPC Administrator as well as written permission obtained to distribute same.

Requests for this TPC Benchmark C Full Disclosure Report should be sent to:

Transaction Processing Performance Council
c/o Shanley Public Relations
777 North First Street, Suite 6000
San Jose, CA 95112-6311

or

Compaq Computer Corporation
Database Performance Engineering
P.O. Box 692000
Houston, TX 77269-2000



PERFORMANCE METRICS INC.
TPC Certified Auditors

February 5, 1999

Mr. Andy Bond
Senior Database Performance Engineer
Compaq Computer Corporation
20555 SH 249
Houston, TX 77070

I have verified the TPC Benchmark™ C client/server for the following configuration:

Platform: ProLiant 7000 6/450 - 2S
Database Manager: Microsoft SQL Server 7.0 Enterprise Edition
Operating System: Microsoft NT version 4.0 Enterprise Edition
Transaction Monitor: Compaq NonStop Tuxedo 1.3

Server: ProLiant 7000				
CPU's	Memory	Disks	90% Response	TpmC
4 Pentium II @ 450 Mhz	Main: 4 GB Cache: 2 MB	182 @ 9.1GB 1 @ 4.1 GB	1.11 sec	22,478.90
4 Client: ProLiant 1600				
2 Pentium II @ 400 Mhz	Main: 384 MB Cache: 512K	1 @ 4.1GB	na	Na

In my opinion, these performance results were produced in compliance with the TPC requirements for the benchmark. The following attributes of the benchmark were given special attention:

- The transactions were correctly implemented.
- The database files were properly sized and populated.
- The database was properly scaled with 1830 warehouses.
- The ACID properties were successfully demonstrated.
- Input data was generated according to the specified percentages.
- Eight hours of mirrored log space was present on the tested system.

2229 Benita Dr. Suite 101, Rancho Cordova, CA 95670
(916) 635-2822 fax: (916) 858-0109 email: Lorna@PerfMetrics.com

Page 1

PERFORMANCE METRICS INC.
TPC Certified Auditors

- Eight hours of growth space for the dynamic tables was present on the tested system.
- The data for the 180 day space calculation was verified.
- The controller cache was disabled on all log disks.
- The steady state portion of the test was 20 minutes.
- One checkpoint was taken before the measured interval.
- One checkpoint was taken during the measured interval.
- The checkpoints were verified to be clear of the guard zone.
- The system pricing was checked for major components and maintenance.
- Third party quotes were verified for compliance.

Auditor Notes:

None

Sincerely,



Lorna Livingtree
Auditor

2229 Benita Dr. Suite 101, Rancho Cordova, CA 95670
(916) 635-2822 fax: (916) 858-0109 email: Lorna@PerfMetrics.com

Page 2

Appendix A: Source Code

The client source code is listed below.

mix

```
mon1, ./norate null mon1.1 ./no1.log
mon2@r2, ./norate null mon2.1 ./no2.log
mon3@r3, ./norate null mon3.1 ./no3.log
mon4@r4, ./norate null mon4.1 ./no4.log
mon5@r5, ./norate null mon5.1 ./no5.log
mon6@r6, ./norate null mon6.1 ./no6.log
mon7@r7, ./norate null mon7.1 ./no7.log
mon8@r8, ./norate null mon8.1 ./no8.log
mon9@r9, ./norate null mon9.1 ./no9.log
mon10@r10, ./norate null mon10.1 ./no10.log
mon11@r11, ./norate null mon11.1 ./no11.log
mon12@r12, ./norate null mon12.1 ./no12.log
mon13@r21, ./norate null mon13.1 ./no13.log
mon14@r14, ./norate null mon14.1 ./no14.log
mon15@r15, ./norate null mon15.1 ./no15.log
mon16@r16, ./norate null mon16.1 ./no16.log
mon17@r17, ./norate null mon17.1 ./no17.log
mon18@r23, ./norate null mon18.1 ./no18.log
mon19@r19, ./norate null mon19.1 ./no19.log
mon20@r20, ./norate null mon20.1 ./no20.log
user0001, ./tpcc -l 3 log/u0001.1 client tigger 1 1 1830
user0002, ./tpcc -l 3 log/u0002.1 client tigger 1 2 1830
user0003, ./tpcc -l 3 log/u0003.1 client tigger 1 3 1830
user0004, ./tpcc -l 1 log/u0004.1 client tigger 1 4 1830
user0005, ./tpcc -l 1 log/u0005.1 client tigger 1 5 1830
.
.
user1000@r2, ./tpcc -l 1 log/u1000.1 client tigger 100 10 1830
user1001@r2, ./tpcc -l 1 log/u1001.1 client tigger 101 1 1830
user1002@r2, ./tpcc -l 1 log/u1002.1 client tigger 101 2 1830
user1003@r2, ./tpcc -l 1 log/u1003.1 client tigger 101 3 1830
user1004@r2, ./tpcc -l 1 log/u1004.1 client tigger 101 4 1830
user1005@r2, ./tpcc -l 1 log/u1005.1 client tigger 101 5 1830
.
.
user2000@r3, ./tpcc -l 1 log/u2000.1 client tigger 200 10 1830
user2001@r3, ./tpcc -l 1 log/u2001.1 client tigger 201 1 1830
user2002@r3, ./tpcc -l 1 log/u2002.1 client tigger 201 2 1830
user2003@r3, ./tpcc -l 1 log/u2003.1 client tigger 201 3 1830
user2004@r3, ./tpcc -l 1 log/u2004.1 client tigger 201 4 1830
user2005@r3, ./tpcc -l 1 log/u2005.1 client tigger 201 5 1830
.
.
user3000@r4, ./tpcc -l 1 log/u3000.1 client tigger 300 10 1830
```

```
user3001@r4, ./tpcc -l 1 log/u3001.1 client tigger 301 1 1830
user3002@r4, ./tpcc -l 1 log/u3002.1 client tigger 301 2 1830
user3003@r4, ./tpcc -l 1 log/u3003.1 client tigger 301 3 1830
user3004@r4, ./tpcc -l 1 log/u3004.1 client tigger 301 4 1830
user3005@r4, ./tpcc -l 1 log/u3005.1 client tigger 301 5 1830
.
.
user18000@r20, ./tpcc -l 1 log/u18000.1 client tigger 1800 10 1830
user18001@r20, ./tpcc -l 1 log/u18001.1 client tigger 1801 1 1830
user18002@r20, ./tpcc -l 1 log/u18002.1 client tigger 1801 2 1830
user18003@r20, ./tpcc -l 1 log/u18003.1 client tigger 1801 3 1830
user18004@r20, ./tpcc -l 1 log/u18004.1 client tigger 1801 4 1830
user18005@r20, ./tpcc -l 1 log/u18005.1 client tigger 1801 5 1830
.
.
user18296@r20, ./tpcc -l 1 log/u18296.1 client tigger 1830 6 1830
user18297@r20, ./tpcc -l 1 log/u18297.1 client tigger 1830 7 1830
user18298@r20, ./tpcc -l 1 log/u18298.1 client tigger 1830 8 1830
user18299@r20, ./tpcc -l 1 log/u18299.1 client tigger 1830 9 1830
user18300@r20, ./tpcc -l 1 log/u18300.1 client tigger 1830 10 1830
```

makefile.client

```
!IF "$(CFG)" == ""
CFG=Release
!MESSAGE No configuration specified. Defaulting to Debug
!ENDIF

!IF "$(SQL_LOC)" == ""
SQL_LOC=C:\prg_tool.kit\i386
!MESSAGE No SQL_LOC specified. Defaulting to C:\prg_tool.kit\i386
!ENDIF

!IF "$(CFG)" != "Release" && "$(CFG)" != "Debug"
!MESSAGE Invalid configuration "$(CFG)" specified.
!MESSAGE You can specify a configuration when running NMAKE on this makefile
!MESSAGE by defining the macro CFG on the command line. For example:
!MESSAGE
!MESSAGE NMAKE CFG="Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "Release"
!MESSAGE "Debug"
!MESSAGE
!ERROR An invalid configuration is specified.
!ENDIF

OUTDIR          = .
SRCDIR          = .\Src
OBJDIR          = .\Objs
OUTDIR          = .\Bin
DBLIB           = $(SQL_LOC)
DBLIBINC        = $(DBLIB)
DBLIBBDIR       = $(DBLIB)
TUXDIR          = C:\TUXEDO

!IF "$(CFG)" != "Debug"
LDEBUG         =
CDEBUG         =
```

```

LDEBUG_RG =
CDEBUG_RG =
DEBUG_ =
FLAGS = /D "WIN32" /D "WINDOWS" /D "_TMSTHEADS" /D "LOCAL_ALLOC"
OPT = /G6 /Ox /Gs16384 /E262144
!ELSE
LDEBUG = /debug /pdb:$(OBJDIR)\tpcc.pdb
CDEBUG = /Zi /Yd /Fd$(OBJDIR)\tpcc.pdb
LDEBUG_RG = /debug /pdb:$(OBJDIR)\install.pdb
CDEBUG_RG = /Zi /Yd /Fd$(OBJDIR)\install.pdb
FLAGS = /D "_DEBUG" /D "WIN32" /D "WINDOWS" /D "_TMSTHEADS" /D
"LOCAL_ALLOC"
OPT = /Od
!ENDIF

CFLAGS = /D _TMSTHEADS

LINK32_LIBS = user32.lib msacm32.lib advapi32.lib $(DBLIBDIR)\ntwdblib.lib
TUX_LIBS = $(TUXDIR)\lib\libtux.lib $(TUXDIR)\lib\libbuft.lib
$(TUXDIR)\lib\libtux2.lib \
$(TUXDIR)\lib\libfml.lib $(TUXDIR)\lib\libfml32.lib $(TUXDIR)\lib\libgp.lib
OTHER_LIBS = wsoc32.lib kernel32.lib gdi32.lib comdlg32.lib winspool.lib
LINK32_OBJS = "$(OBJDIR)\tpcc.obj" "$(OBJDIR)\tpcc.res"
LINK32_DEF = "$(SRCDIR)\tpcc.def"
LINK32_FLAGS = /nologo /subsystem:windows /dll /incremental:no $(LDEBUG)
/def:"$(LINK32_DEF)" /out:"$(OUTDIR)\tpcc.dll"

LINK32_LIBS_RG = user32.lib gdi32.lib advapi32.lib version.lib comctl32.lib
LINK32_OBJS_RG = "$(OBJDIR)\install.obj" "$(OBJDIR)\install.res"
LINK32_FLAGS_RG = /nologo /subsystem:windows /incremental:no $(LDEBUG_RG)
/out:$(OUTDIR)\install.exe

ALL: $(OBJDIR)\. $(OUTDIR)\. $(OUTDIR)\tpcc.dll $(OUTDIR)\Neworder.exe \
$(OUTDIR)\Payment.exe $(OUTDIR)\q.exe \
$(OUTDIR)\Stocklevel.exe $(OUTDIR)\Orderstatus.exe $(OUTDIR)\Delivery.exe
$(OUTDIR)\Delirpt.exe

touch:
touch $(SRCDIR)/*.c

$(OBJDIR)\.:
if not exist $(OBJDIR) md $(OBJDIR)

$(OUTDIR)\.:
if not exist $(OUTDIR) md $(OUTDIR)

"$(OBJDIR)\tpcc.obj": "$(SRCDIR)\tpcc.c" "$(SRCDIR)\tpcc.h"
cl.exe /nologo /MT /W3 $(CDEBUG) $(OPT) /I $(DBLIBINC) /I
$(TUXDIR)\include $(FLAGS) /Fo$(OBJDIR)\tpcc.obj /c "$(SRCDIR)\tpcc.c"

$(OBJDIR)\tpcc.res: $(SRCDIR)\tpcc.rc
rc.exe /l 0x409 /fo $(OBJDIR)\tpcc.res $(FLAGS) $(SRCDIR)\tpcc.rc

$(OUTDIR)\tpcc.dll: $(LINK32_OBJS) $(LINK32_DEF)
link.exe $(LINK32_FLAGS) $(LINK32_OBJS) $(LINK32_LIBS) $(TUX_LIBS)
$(OTHER_LIBS)

$(OUTDIR)\q.exe: $(SRCDIR)\q.c
$(TUXDIR)\bin\buildserver -v -k /f $(SRCDIR)\q.c /o $(OUTDIR)\q.exe \
/s Q1,Q2,Q3,Q4,Q5,Q6,Q7,Q8,Q9,Q10 /l "$(LINK32_LIBS) /I $(DBLIBINC)"
del q.obj

$(OUTDIR)\Neworder.exe: $(SRCDIR)\neworder.c

```

```

$(TUXDIR)\bin\buildserver -v -k /f $(SRCDIR)\neworder.c /o
$(OUTDIR)\Neworder.exe \
/s NEWORDER:NEWORDER /l "$(LINK32_LIBS) /I $(DBLIBINC)"
del neworder.obj

$(OUTDIR)\Payment.exe: $(SRCDIR)\payment.c
$(TUXDIR)\bin\buildserver /f $(SRCDIR)\payment.c /o $(OUTDIR)\Payment.exe
\
/s PAYMENT:PAYMENT /l "$(LINK32_LIBS) /I $(DBLIBINC)"
del payment.obj

$(OUTDIR)\Orderstatus.exe: $(SRCDIR)\orderstatus.c
$(TUXDIR)\bin\buildserver /f $(SRCDIR)\orderstatus.c /o
$(OUTDIR)\Orderstatus.exe \
/s ORDERSTATUS:ORDERSTATUS /l "$(LINK32_LIBS) /I $(DBLIBINC)"
del orderstatus.obj

$(OUTDIR)\Stocklevel.exe: $(SRCDIR)\stocklevel.c
$(TUXDIR)\bin\buildserver /f $(SRCDIR)\stocklevel.c /o
$(OUTDIR)\Stocklevel.exe \
/s STOCKLEVEL:STOCKLEVEL /l "$(LINK32_LIBS) /I $(DBLIBINC)"
del stocklevel.obj

$(OUTDIR)\Delivery.exe: $(SRCDIR)\delivery.c
$(TUXDIR)\bin\buildserver /f $(SRCDIR)\delivery.c /o
$(OUTDIR)\Delivery.exe \
/s DELIVERY:DELIVERY /l "$(LINK32_LIBS) /I $(DBLIBINC)"
del delivery.obj

$(OUTDIR)\Delirpt.exe: $(SRCDIR)\delirpt.c
cl.exe $(SRCDIR)\delirpt.c /o "$(OUTDIR)\Delirpt.exe"
del delirpt.obj

```

delirpt.c

```

/* FILE: DELIRPT.C
 * Microsoft TPC-C Kit Ver. 3.00.000
 * Copyright Microsoft, 1996
 *
 * PURPOSE: Delivery report processing application
 * Author: Philip Durr
 * philipdu@Microsoft.com
 */

#include <windows.h>
#include <stdio.h>
#include <stdlib.h>

#define LOGFILE_READ_EOF 0 //check log file flag return current state
#define LOGFILE_CLEAR_EOF 1 //clear end of log file flag
#define LOGFILE_SET_EOF 2 //set flag end of log file reached

#define INTERVAL .01 //90th percentile calculation bucket

interval

#define ERR_SUCCESS 1000 //success no error

```

```

#define ERR_READING_LOGFILE          1001
//io errors ocured reading delivery log file
#define ERR_INSUFFICIENT_MEMORY     1002
//insuficient memory to process 90th percentile report
#define ERR_CANNOT_OPEN_RESULTS_FILE 1005
//Cannot open delivery results file delilog.

typedef struct _RPTLINE
{
    SYSTEMTIME      start;
//delilog report line start time
    SYSTEMTIME      end;
//delilog report line end time
    int             response;
//delilog report line time
    int             delivery_took_in_milliseconds;
    int             w_id;
//delilog report line
    int             warehouse_id_for_delivery;
    int             o_carrier_id;
//delilog report line carier id
    int             for_delivery;
    int             items[10];
//delilog report line
} RPTLINE, *PRPTLINE;

//error message structure used in ErrorMessage API
typedef struct _SERRORMSG
{
    int             iError;
//error id of
    char            message;
    char            szMsg[80];
//message to sent to
    int             browser;
} SERRORMSG;

int             versionMS = 3;
//delirtp version
int             versionMM = 0;
int             versionLS = 2;
int             iReport;
//delirtp report to process
int             iStartTime;
//begin times to accept for report
int             iEndTime;
//end times to accept for report
FILE            *fpLog;
//log file stream

//Local function prototypes
void            main(int argc, char *argv[]);
static int      Init(void);
static void     Restore(void);
static int      DoReport(void);
int             AverageResponse(void);
int             SkippedDelivery(void);
int             Percentile90th(void);
BOOL            CheckTimes(PRPTLINE pRptLine);
static int      OpenLogFile(void);
static void     CloseLogFile(void);
static void     ResetLogFile(void);
static void     LogEOF(int iOperation);
static BOOL     ReadReportLine(char *szBuffer, PRPTLINE pRptLine);

```

```

static BOOL     ParseReportLine(char *szLine, PRPTLINE pRptLine);
static BOOL     ParseDate(char *szDate, LPSYSTEMTIME pTime);
static BOOL     ParseTime(char *szTime, LPSYSTEMTIME pTime);
static void     ErrorMessage(int iError);
static BOOL     GetParameters(int argc, char *argv[]);
static void     PrintParameters(void);
static void     PrintHeader(void);
static void     cls(void);
static BOOL     IsNumeric(char *ptr);

/* FUNCTION: int main(int argc, char *argv[])
 *
 * PURPOSE:      This function is the beginning execution point for the delivery
 * executable.
 *
 * ARGUMENTS:    int             argc             number of command line arguments
 *               char            *argv[]         array of command line
 *               argument pointers
 *
 * RETURNS:      None
 *
 * COMMENTS:     None
 */

void main(int argc, char *argv[])
{
    int             iError;
    PrintHeader();
    if ( GetParameters(argc, argv) )
    {
        PrintParameters();
        return;
    }
    if ( (iError=Init()) != ERR_SUCCESS )
    {
        ErrorMessage(iError);
        Restore();
        return;
    }
    if ( (iError = DoReport()) != ERR_SUCCESS )
        ErrorMessage(iError);
    Restore();
    return;
}

/* FUNCTION: static int Init(void)
 *
 * PURPOSE:      This function initializes the delirtp application.
 *
 * ARGUMENTS:    None
 *
 * RETURNS:      None
 *
 * COMMENTS:     None
 */

```

```

*
*/
static int Init(void)
{
    int iError;

    if ( (iError = OpenLogFile()) )
        return iError;
    return TRUE;
}

/* FUNCTION: static void Restore(void)
*
* PURPOSE:      This function cleans up the delirpt application before
termination.
*
* ARGUMENTS:    None
*
* RETURNS:      None
*
* COMMENTS:     None
*/
static void Restore(void)
{
    CloseLogFile();
    return;
}

/* FUNCTION: static int DoReport(void)
*
* PURPOSE:      This function dispatches the requested report.
*
* ARGUMENTS:    None
*
* RETURNS:      ERR_SUCCESS if successfull or error code if an error
occurs.
*
* COMMENTS:     None
*/
static int DoReport(void)
{
    int iRc;

    switch(iReport)
    {
        case 1:
            iRc = AverageResponse();
            break;
        case 2:
            iRc = Percentile90th();
            break;
        case 3:
            iRc = SkippedDelivery();
            break;
        case 4:
            if ( (iRc = AverageResponse()) != ERR_SUCCESS )
                break;
            if ( (iRc = Percentile90th()) != ERR_SUCCESS )

```

```

                break;
            if ( (iRc = SkippedDelivery()) != ERR_SUCCESS )
                break;
        }
        return iRc;
    }

/* FUNCTION: int AverageResponse(void)
*
* PURPOSE:      This function processes the AverageResponse report.
*
* ARGUMENTS:    None
*
* RETURNS:      ERR_SUCCESS if successfull or error code if an error
occurs.
*
* COMMENTS:     None
*/
int AverageResponse(void)
{
    RPTLINE reportLine;
    int iTotalResponse;
    int iLines;
    double fAverage;
    char szDelivery[128];

    ResetLogFile();

    iTotalResponse = 0;
    iLines = 0;
    printf("\n\n***** Average Response Time Report *****\n");
    while ( !LogEOF(LOGFILE_READ_EOF) )
    {
        if ( ReadReportLine(szDelivery, &reportLine) )
            return ERR_READING_LOGFILE;
        if ( szDelivery[0] == '*' )
            continue;
        if ( !LogEOF(LOGFILE_READ_EOF) )
        {
            if ( CheckTimes(&reportLine) )
                continue;
            iLines++;
            iTotalResponse += reportLine.response;

            if ( iLines % 10 == 0 )
                printf("Reading Report Line:\t%d\r",
                    iLines);
        }
    }
    printf("                                \r");
    if ( iLines == 0 )
    {
        printf("No deliveries found.\n");
    }
    else
    {
        fAverage = ((double)iTotalResponse /
            (double)iLines)/(double)1000;
        printf("Total Deliveries:      %10.0f\n", (float)iLines);
    }
}

```

```

        printf("Total Response Times: %10.3f\n",
((float)iTotalResponse/(float)1000));
        printf("Average Response Time: %10.3f\n", fAverage);
    }

    return ERR_SUCCESS;
}

/* FUNCTION: int Percentile90th(void)
 *
 * PURPOSE:          This function processes the 90th percentile report.
 *
 * ARGUMENTS:       None
 *
 * RETURNS:         ERR_SUCCESS if successfull or error code if an error
occurs.
 *
 * COMMENTS:        This function requires enough space to allocate needed
 *                  buckets which will be 2 * max response time
in
 *                  deci-seconds.
 *
 */

int Percentile90th(void)
{
    RPTLINE reportLine;
    int      iBucketSize;
    int      i;
    int      iResponseSeconds;
    int      iMaxSeconds;
    int      iTotalsBuckets;
    double   iTotal;
    double   i90thPercent;
    short    *psBuckets;
    char     szDelivery[128];

    printf("\n\n***** 90th Percentile *****\n");
    printf("Calculating Max Response Seconds...\n");

    ResetLogFile();

    iMaxSeconds = -1;
    while ( !LogEOF(LOGFILE_READ_EOF) )
    {
        if ( ReadReportLine(szDelivery, &reportLine) )
            return ERR_READING_LOGFILE;
        if ( szDelivery[0] == '*' )
            continue;
        if ( !LogEOF(LOGFILE_READ_EOF) )
        {
            if ( iMaxSeconds < reportLine.response )
                iMaxSeconds = reportLine.response;
        }
    }

    iTotalBuckets = iMaxSeconds + 1;

    printf("Allocating Buckets...\n");

    iBucketSize = iTotalBuckets * sizeof(short);

    if ( !(psBuckets = (short *)malloc(iBucketSize)) )

```

```

        return ERR_INSUFFICIENT_MEMORY;

    ZeroMemory(psBuckets, iBucketSize);

    iTotal = 0;

    ResetLogFile();
    printf("Calculating Distribution...\n");

    while ( !LogEOF(LOGFILE_READ_EOF) )
    {
        if ( ReadReportLine(szDelivery, &reportLine) )
            return ERR_READING_LOGFILE;
        if ( szDelivery[0] == '*' )
            continue;
        if ( !LogEOF(LOGFILE_READ_EOF) )
        {
            if ( CheckTimes(&reportLine) )
                continue;
            psBuckets[reportLine.response]++;
            iTotal++;
        }
    }

    i90thPercent = iTotal * .9;

    for(i=0, iTotal = 0.0; iTotal < i90thPercent; iTotal +=
(double)psBuckets[i] )
        i++;

    printf("90th Percentile = %d.%d\n", i/1000, (i % 1000));

    free(psBuckets);

    return ERR_SUCCESS;
}

/* FUNCTION: int SkippedDelivery(void)
 *
 * PURPOSE:          This function processes the Skipped Deliveries report.
 *
 * ARGUMENTS:       None
 *
 * RETURNS:         ERR_SUCCESS if successfull or error code if an error
occurs.
 *
 * COMMENTS:        None
 *
 */

int SkippedDelivery(void)
{
    RPTLINE reportLine;
    char     szDelivery[128];
    int      i;
    int      items[10];

    ResetLogFile();

    printf("\n\n***** Skipped Delivery Report *****\n");
    memset(items, 0, sizeof(items));
    printf("Reading Delivery Log File...");

```



```

while ( !LogEOF(LOGFILE_READ_EOF) )
{
    if ( ReadReportLine(szDelivery, &reportLine) )
        return ERR_READING_LOGFILE;
    if ( szDelivery[0] == '*' )
        continue;
    if ( !LogEOF(LOGFILE_READ_EOF) )
    {
        if ( CheckTimes(&reportLine) )
            continue;
        for(i=0; i<10; i++)
        {
            if ( !reportLine.items[i] )
                items[i]++;
        }
    }
    printf("\n");
    printf("Skipped delivery table.\n");
    printf(" 1   2   3   4   5   6   7   8   9  10 \n");
    printf("-----\n");
    for(i=0; i<10; i++)
        printf("%4.4d ", items[i]);
    printf("\n");

    return ERR_SUCCESS;
}

/* FUNCTION: BOOL CheckTimes(PRPTLINE pRptLine)
 *
 * PURPOSE:      This function checks to see if the delilog record falls within
the
 *              begin and end time from the command line.
 *
 * ARGUMENTS:    PRPTLINE  pRptLine  delilog processed report line.
 *
 * RETURNS:      BOOL      FALSE     if report line is not within the
 *              TRUE      if the report
line is within the
 *              requested start and end times.
 *
 * COMMENTS:     If startTime and endTime are both 0 then the user requested
 *              the default behavior which is all records in
delilog are
 *              valid.
 */

BOOL CheckTimes(PRPTLINE pRptLine)
{
    int      iRptEndTime;
    int      iRptStartTime;

    iRptStartTime = (pRptLine->start.wHour * 3600000) + (pRptLine->start.wMinute * 60000) + (pRptLine->start.wSecond * 1000) + pRptLine->start.wMilliseconds;
    iRptEndTime = (pRptLine->end.wHour * 3600000) + (pRptLine->end.wMinute * 60000) + (pRptLine->end.wSecond * 1000) + pRptLine->end.wMilliseconds;

    if ( iStartTime == 0 && iEndTime == 0 )
        return FALSE;
}

```

```

    if ( iStartTime <= iRptStartTime && iEndTime >= iRptEndTime )
        return FALSE;

    return TRUE;
}

/* FUNCTION: int OpenLogFile(void)
 *
 * PURPOSE:      This function opens the delivery log file for use.
 *
 * ARGUMENTS:    None
 *
 * RETURNS:      int      ERR_CANNOT_OPEN_RESULTS_FILE  Cannot create
results log file.
 *              ERR_SUCCESS  Log file successfully opened
 *
 * COMMENTS:     None
 */

static int OpenLogFile(void)
{
    fpLog = fopen("delilog.", "rb");

    if ( !fpLog )
        return ERR_CANNOT_OPEN_RESULTS_FILE;

    return ERR_SUCCESS;
}

/* FUNCTION: int CloseLogFile(void)
 *
 * PURPOSE:      This function closes the delivery log file.
 *
 * ARGUMENTS:    None
 *
 * RETURNS:      None
 *
 * COMMENTS:     None
 */

static void CloseLogFile(void)
{
    if ( fpLog )
        fclose(fpLog);

    return;
}

/* FUNCTION: static void ResetLogFile(void)
 *
 * PURPOSE:      This function prepares the delilog. file for reading
 *
 * ARGUMENTS:    None
 *
 * RETURNS:      None
 *
 * COMMENTS:     None
 */

```

```

*
*/
static void ResetLogFile(void)
{
    fseek(fpLog, 0L, SEEK_SET);
    LogEOF(LOGFILE_CLEAR_EOF);

    return;
}

/* FUNCTION: static BOOL LogEOF(int iOperation)
*
* PURPOSE:      This function tracks and reports the end of file condition
*               on the delilog file.
*
* ARGUMENTS:   int iOperation      requested operation this can be:
*
* LOGFILE_READ_EOF   check log file flag return current state
*
* LOGFILE_CLEAR_EOF  clear end of log file flag
*
* LOGFILE_SET_EOF    set flag end of log file reached
*
* RETURNS:      None
*
* COMMENTS:    None
*/
static BOOL LogEOF(int iOperation)
{
    static BOOL bEOF;

    switch(iOperation)
    {
        case LOGFILE_READ_EOF:
            return bEOF;
            break;
        case LOGFILE_CLEAR_EOF:
            bEOF = FALSE;
            break;
        case LOGFILE_SET_EOF:
            bEOF = TRUE;
            break;
    }
    return FALSE;
}

/* FUNCTION: static BOOL ReadReportLine(char *szBuffer, PRPTLINE pRptLine)
*
* PURPOSE:      This function reads a text line from the delilog file.
*               on the delilog file.
*
* ARGUMENTS:   char *szBuffer buffer to placed read delilog file
line into.
*
*               PRPTLINE pRptLine returned structure
containing parsed delilog
*
*               report line.
*
* RETURNS:     FALSE if successfull or TRUE if an error occurs.

```

```

*
* COMMENTS:    None
*/
static BOOL ReadReportLine(char *szBuffer, PRPTLINE pRptLine)
{
    int i = 0;
    int ch;
    int iEof;

    while( i < 128 )
    {
        ch = fgetc(fpLog);
        if ( iEof = feof(fpLog) )
            break;
        if ( ch == '\r' )
        {
            if ( i )
                break;
            continue;
        }
        if ( ch == '\n' )
            continue;
        szBuffer[i++] = ch;
    }

    //delivery item format is to long cannot be a valid delivery item
    if ( i >= 128 )
        return TRUE;

    szBuffer[i] = 0;
    if ( iEof )
    {
        LogEOF(LOGFILE_SET_EOF);
        if ( i == 0 )
            return FALSE;
    }
    if ( szBuffer[0] == '*' )
    {
        //error line ignore
        return FALSE;
    }
    return ParseReportLine(szBuffer, pRptLine);
}

/* FUNCTION: static BOOL ParseReportLine(char *szLine, PRPTLINE pRptLine)
*
* PURPOSE:      This function reads a text line from the delilog file.
*               on the delilog file.
*
* ARGUMENTS:   char *szLine      buffer containing the
delilog file line to be parsed.
*
*               PRPTLINE pRptLine returned structure
containing parsed delilog
*
*               report line values.
*
* RETURNS:     FALSE if successfull or TRUE if an error occurs.
*
* COMMENTS:    None

```

```

*/
static BOOL ParseReportLine(char *szLine, PRPTLINE pRptLine)
{
    int i;

    if ( ParseDate(szLine, &pRptLine->start) )
        return TRUE;

    pRptLine->end.wYear = pRptLine->start.wYear;
    pRptLine->end.wMonth = pRptLine->start.wMonth;
    pRptLine->end.wDay = pRptLine->start.wDay;

    if ( !(szLine = strchr(szLine, ',')) )
        return TRUE;
    szLine++;

    if ( ParseTime(szLine, &pRptLine->start) )
        return TRUE;

    if ( !(szLine = strchr(szLine, ',')) )
        return TRUE;
    szLine++;

    if ( ParseTime(szLine, &pRptLine->end) )
        return TRUE;

    if ( !(szLine = strchr(szLine, ',')) )
        return TRUE;
    szLine++;

    if ( !IsNumeric(szLine) )
        return TRUE;
    pRptLine->response = atoi(szLine);

    if ( !(szLine = strchr(szLine, ',')) )
        return TRUE;
    szLine++;

    if ( !IsNumeric(szLine) )
        return TRUE;
    pRptLine->w_id = atoi(szLine);

    if ( !(szLine = strchr(szLine, ',')) )
        return TRUE;
    szLine++;

    if ( !IsNumeric(szLine) )
        return TRUE;
    pRptLine->o_carrier_id = atoi(szLine);

    if ( !(szLine = strchr(szLine, ',')) )
        return TRUE;
    szLine++;

    for(i=0; i<10; i++)
    {
        if ( !IsNumeric(szLine) )
            return TRUE;
        pRptLine->items[i] = atoi(szLine);

        if ( i<9 && !(szLine = strchr(szLine, ',')) )
            return TRUE;
    }
}

```

```

        szLine++;
    }
    return FALSE;
}

/* FUNCTION: static BOOL ParseDate(char *szDate, LPSYSTEMTIME pTime)
*
* PURPOSE:      This function validates and extracts a date string in the format
*               yy/mm/dd into an SYSTEMTIME structure.
*
* ARGUMENTS:    char          *szDate          buffer
                containing the date to be parsed.
                *               LPSYSTEMTIME    pTime
                system time structure where date will be placed.
*
* RETURNS:      FALSE if successfull or TRUE if an error occurs.
*
* COMMENTS:     None
*/

static BOOL ParseDate(char *szDate, LPSYSTEMTIME pTime)
{
    if ( !isdigit(*szDate) || !isdigit(*(szDate+1)) || *(szDate+2) != '/' ||
        !isdigit(*(szDate+3)) || !isdigit(*(szDate+4)) || *(szDate+5)
!= '/' ||
        !isdigit(*(szDate+6)) || !isdigit(*(szDate+7)) )
        return TRUE;

    pTime->wYear = atoi(szDate);
    pTime->wMonth = atoi(szDate+3);
    pTime->wDay = atoi(szDate+6);

    if ( pTime->wMonth > 12 || pTime->wMonth < 0 || pTime->wDay > 31 || pTime-
>wDay < 0 )
        return TRUE;

    return FALSE;
}

/* FUNCTION: static BOOL ParseTime(char *szTime, LPSYSTEMTIME pTime)
*
* PURPOSE:      This function validates and extracts a time string in the format
*               hh:mm:ss:mmm into an SYSTEMTIME structure.
*
* ARGUMENTS:    char          *szTime          buffer
                containing the time to be parsed.
                *               LPSYSTEMTIME    pTime
                system time structure where date will be placed.
*
* RETURNS:      FALSE if successfull or TRUE if an error occurs.
*
* COMMENTS:     None
*/

static BOOL ParseTime(char *szTime, LPSYSTEMTIME pTime)
{
    if ( !isdigit(*szTime) || !isdigit(*(szTime+1)) || *(szTime+2) != ':' ||

```

```

!isdigit(*(szTime+3)) || !isdigit(*(szTime+4)) || *(szTime+5) !=
': ' ||
!isdigit(*(szTime+6)) || !isdigit(*(szTime+7)) || *(szTime+8)
!= ': ' ||
!isdigit(*(szTime+9)) || !isdigit(*(szTime+10)) ||
!isdigit(*(szTime+11)) )
return TRUE;

pTime->wHour = atoi(szTime);
pTime->wMinute = atoi(szTime+3);
pTime->wSecond = atoi(szTime+6);
pTime->wMilliseconds = atoi(szTime+9);

if ( pTime->wHour > 23 || pTime->wHour < 0 ||
    pTime->wMinute > 59 || pTime->wMinute < 0 ||
    pTime->wSecond > 59 || pTime->wSecond < 0 ||
    pTime->wMilliseconds < 0 )
return TRUE;

if ( pTime->wMilliseconds > 999 )
{
    pTime->wSecond += (pTime->wMilliseconds/1000);
    pTime->wMilliseconds = pTime->wMilliseconds % 1000;
}

return FALSE;
}

/* FUNCTION: void ErrorMessage(int iError)
 *
 * PURPOSE: This function displays an error message in the delivery
executable's console window.
 *
 * ARGUMENTS: int iError error id to be displayed
 *
 * RETURNS: None
 *
 * COMMENTS: None
 */

static void ErrorMessage(int iError)
{
    int i;

    static SERRORMSG errorMsgs[] =
    {
        { ERR_SUCCESS,
          "Success, no error."
        },
        { ERR_CANNOT_OPEN_RESULTS_FILE,
          "Cannot open delivery results file delilog."
        },
        { ERR_READING_LOGFILE,
          "Reading delivery log file, Delivery item format incorrect."
        },
        { ERR_INSUFFICIENT_MEMORY,
          "insufficient memory to process 90th percentile report."
        },
        { 0,
          ""
        }
    };
};

```

```

for(i=0; errorMsgs[i].szMsg[0]; i++)
{
    if ( iError == errorMsgs[i].iError )
    {
        printf("\nError(%d): %s", iError, errorMsgs[i].szMsg);
        return;
    }
}
printf("Error(%d): %s", errorMsgs[0].szMsg);
return;
}

/* FUNCTION: BOOL GetParameters(int argc, char *argv[])
 *
 * PURPOSE: This function parses the command line passed in to the delivery
executable, initializing
and filling in global variable parameters.
 *
 * ARGUMENTS: int argc number of command line arguments
passed to delivery
char *argv[] array of command line
argument pointers
 *
 * RETURNS: BOOL FALSE parameter read successful
TRUE user has
requested parameter information screen be displayed.
 *
 * COMMENTS: None
 */

static BOOL GetParameters(int argc, char *argv[])
{
    int i;
    SYSTEMTIME startTime;
    SYSTEMTIME endTime;

    iStartTime = 0;
    iEndTime = 0;
    iReport = 4;

    for(i=0; i<argc; i++)
    {
        if ( argv[i][0] == '-' || argv[i][0] == '/' )
        {
            switch(argv[i][1])
            {
                case 'S':
                case 's':
                    if ( ParseTime(argv[i]+2,
                                &startTime) )
                        return TRUE;
                    iStartTime = (startTime.wHour *
3600000) + (startTime.wMinute * 60000) + (startTime.wSecond * 1000) +
startTime.wMilliseconds;
                    break;
                case 'E':
                case 'e':
                    if ( ParseTime(argv[i]+2,
                                &endTime) )
                        return TRUE;

```

```

        iEndTime = (endTime.wHour *
3600000) + (endTime.wMinute * 60000) + (endTime.wSecond * 1000) +
        endTime.wMilliseconds;
        break;
        case 'R':
        case 'r':
            iReport = atoi(argv[i]+2);
            if ( iReport > 4 || iReport < 1 )
                iReport = 4;
            break;
        case '?':
            return TRUE;
        }
    }
    return FALSE;
}

/* FUNCTION: void PrintParameters(void)
 * PURPOSE:      This function displays the supported command line flags.
 * ARGUMENTS:   None
 * RETURNS:     None
 * COMMENTS:    None
 */

static void PrintParameters(void)
{
    PrintHeader();
    printf("DELIRPT:\n\n");
    printf("Parameter
Default\n");
    printf("-----\n");
    printf("-S Start Time HH:MM:SS:MMM
\n");
    printf("-E End Time HH:MM:SS:MMM
\n");
    printf("-R 1)Average Response, 2)90th 3) Skipped 4) All
\n");
    printf("-? This help screen\n\n");
    printf("Note: Command line switches are NOT case sensitive.\n");

    return;
}

/* FUNCTION: void PrintHeader(void)
 * PURPOSE:      This function displays the delivery report applications banner
information.
 * ARGUMENTS:   None
 * RETURNS:     None
 * COMMENTS:    None
 */

```

```

static void PrintHeader(void)
{
    cls();

    printf("*****\n");
    printf("                *\n");
    printf(" Microsoft SQL Server 6.5                *\n");
    printf("                *\n");
    printf(" HTML TPC-C BENCHMARK KIT: Delivery Report *\n");
    printf(" Version %d.%2.2d.%3.3d                *\n",
versionMS, versionMM, versionLS);
    printf("                *\n");
    printf("*****\n\n");

    return;
}

/* FUNCTION: void cls(void)
 * PURPOSE:      This function clears the console window
 * ARGUMENTS:   None
 * RETURNS:     None
 * COMMENTS:    None
 */

static void cls(void)
{
    HANDLE hConsole;
    COORD coordScreen = { 0, 0 }; //here's where
we'll home the cursor
    DWORD cCharsWritten;
    CONSOLE_SCREEN_BUFFER_INFO csbi; //to get buffer info
    DWORD dwConSize; //number of character cells in the current buffer

    hConsole = GetStdHandle(STD_OUTPUT_HANDLE);

    //get the number of character cells in the current buffer

    GetConsoleScreenBufferInfo( hConsole, &csbi );
    dwConSize = csbi.dwSize.X * csbi.dwSize.Y;

    //fill the entire screen with blanks
    FillConsoleOutputCharacter( hConsole, (TCHAR) ' ', dwConSize, coordScreen,
&cCharsWritten );
    GetConsoleScreenBufferInfo( hConsole, &csbi );

    //now set the buffer's attributes accordingly
    FillConsoleOutputAttribute( hConsole, csbi.wAttributes,dwConSize,
coordScreen, &cCharsWritten );

    //put the cursor at (0, 0)
    SetConsoleCursorPosition( hConsole, coordScreen );

    return;
}

/* FUNCTION: BOOL IsNumeric(char *ptr)
 *

```

```

* PURPOSE:      This function determines if a string is numeric. It fails if any
characters other
*              than numeric and null terminator are present.
*
* ARGUMENTS:   char          *ptr      pointer to string to
check.
*
* RETURNS:     BOOL          FALSE    if string is not all numeric
*              TRUE          if string
contains only numeric characters i.e. '0' - '9'
*
* COMMENTS:    A comma is counted as a valid delimiter.
*/

static BOOL IsNumeric(char *ptr)
{
    if ( *ptr == 0 )
        return FALSE;

    while( *ptr && isdigit(*ptr) )
        ptr++;
    if ( !*ptr || *ptr == ',' )
        return TRUE;
    else
        return FALSE;
}

```

delivery.c

```

/* FILE:      DELIVERY.C
*
* Based on: Microsoft TPC-C Kit Ver. 3.00.000
*
* Copyright Microsoft, 1996
* Copyright Performance Tuning Corporation,
1997
*
* PURPOSE:   New Order Tuxedo Server.
* Author:    Philip Durr
*            philipdu@microsoft.com
*
* MODIFIED   Changed for modularity and to allow for the Tuxedo TM
*
* Author:    Edward Whalen
*            Performance Tuning Corporation
*            ewhalen@perftuning.com
*/

#include <windows.h>
#include <process.h>
#include <stdio.h>
#include <stdarg.h>
#include <malloc.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <io.h>

#define DBNTWIN32
#include <sqlfront.h>

```

```

#include <sqldb.h>

#include "trans.h" //tpckit
transaction header contains definations of structures specific to TPC-C
#include "httpext.h" //ISAPI DLL
information header

#include "tpcc.h" //this dlls
specific structure, value e.t. header.

#include <tmenv.h>
#include <xa.h>
#include <atmi.h>

BOOL      bLog          = FALSE;
BOOL      bFlush        = FALSE;
          //Flush delivery log info when written.
BOOL      verbose       = FALSE;
BOOL      bError        = FALSE;

int        iThreads     = 5;
int        iMaxWareHouses = 500;
int        iDelayMs     = 100;
short      iMaxConnections = (short)1;
short      iDeadlockRetry = (short)3;

DBPROCESS *pdbproc;

char          szServer[32];
          //SQL server name
char          szDatabase[32];
          //tpcc database name
char          szUser[32];
          //user name
char          szPassword[32];
          //user password

int spid;

TERM Term;

static char  szTpccLogPath[256]; //path to html log file if logging turned on
in registry.
static char  szErrorLogPath[256]; //path to error log file.

static CRITICAL_SECTION CriticalSection;
static CRITICAL_SECTION ErrorLogCriticalSection;
static EXTENSION_CONTROL_BLOCK *gpECB = NULL;
static int bTpccExit;
          //exit delivery disconnect loop as dll exiting.

extern void TMLog();
extern BOOL SQLInit();
extern void UtilStrCpy();
extern void UtilStrCpy();
extern BOOL SQLOpenConnection();
extern BOOL SQLCloseConnection();

//BOOL      bDone;
          //delivery executable termination request

flag
BOOL      bFlush;
          //Flush delivery log info when written.

```

```

#define ERR_CANNOT_CREATE_THREAD          1000    //Cannot create thread.
#define ERR_DBGGETDATA_FAILED            1001    //Get data failed.
#define ERR_REGISTRY_NOT_SETUP           1002    //Registry not
setup for tpcc.
#define ERR_CANNOT_ACCESS_DELIVERY_FN 1003    //Cannot access ReadDelivery
cache.
#define ERR_CANNOT_ACCESS_REGISTRY       1004    //Cannot access registry
key TPCC.
#define ERR_CANNOT_CREATE_RESULTS_FILE   1005    //Cannot create results
file.

FILE      *fpLog;

/* FUNCTION: tpsvrinit ( int argc, char *argv[] )
 *
 * PURPOSE:      Initialize the Server to Database connection.
 *
 * RETURNS:      int      0      Success
 *              -1      Failure
 *
 * COMMENTS:     None
 */

int tpsvrinit ( int argc, char *argv[] )
{
    if ( GetParameters(argc, argv) )
    {
        PrintParameters();
        return -1;
    }

    if ( verbose )
        TMLog("TPSVRINIT: Delivery: Server %s, Database %s, User %s,
Password %s, Flush %d.",
            szServer, szDatabase, szUser, szPassword, bFlush);

    if ( ! SQLInit() )
    {
        TMLog( "DELIVERY: SQLInit Failed" );
        return -1;
    }

    if ( SQLOpenConnection ( NULL, 0, 0, &pdbproc, szServer, szDatabase,
szUser, szPassword, szDatabase, &spId) )
    {
        TMLog ( "DELIVERY: SQLOpenConnection Failed" );
        dbexit();
        return -1;
    }

    OpenLogFile();

    return 0;
}

/* FUNCTION: tpsvrdone ( void )
 *
 * PURPOSE:      Initialize the Server to Database connection.
 */

```

```

 * RETURNS:      int      0      Success
 *              -1      Failure
 *
 * COMMENTS:     None
 */

void tpsvrdone ( void )
{
    SQLCloseConnection( NULL, pdbproc);
    dbexit();
}

/* FUNCTION: DELIVERY ( TPSVCINFO *rqst )
 *
 * PURPOSE:      Process a New Order request.
 *
 * RETURNS:      int      0      Success
 *              -1      Failure
 *
 * COMMENTS:     None
 */

void DELIVERY ( TPSVCINFO *rqst )
{
    PECBINFO pECBInfo = dbgetuserdata(pdbproc);
    DELIVERY_DATA *dp;

    int size = rqst->len;

    if (verbose)
        TMLog(" DELIVERY: Begin transaction");

    dp = (DELIVERY_DATA *) rqst->data ;

    if (verbose )
    {
        TMLog(" DELIVERY: w_id %d ", dp->w_id);
        TMLog(" DELIVERY: d_id %d ", dp->o_carrier_id);
    }

    bError = FALSE;

    dp->retval = SQLDelivery( pdbproc, dp, iDeadlockRetry);

    if (bError == TRUE)
        dp->retval = -1;

    tpreturn( TFSUCCESS, 0, rqst->data, size, 0);
}

/* FUNCTION: void CalculateElapsedTime(int *pElapsed, LPSYSTEMTIME lpBegin,
LPSYSTEMTIME lpEnd)
 *
 * PURPOSE:      This function calculates the elapsed time a delivery transaction
took.
 *
 * ARGUMENTS:    int      *pElapsed pointer to int
variable to receive calculated elapsed

```



```

        pDelivery->o_id[4], pDelivery->o_id[5], pDelivery->o_id[6],
pDelivery->o_id[7],
        pDelivery->o_id[8], pDelivery->o_id[9] );

        if ( bFlush )
            fflush(fpLog);

        return ERR_SUCCESS;
    }

/* FUNCTION: BOOL SQLDetectDeadlock(DBPROCESS *dbproc)
 *
 * PURPOSE:      This function is used to check for deadlock conditions.
 *
 * ARGUMENTS:    DBPROCESS      *dbproc          DBPROCESS to check
 *
 * RETURNS:      BOOL          FALSE              No
lock condition present
 *              TRUE
 *              Lock condition detected
 *
 * COMMENTS:     None
 *
 */

static BOOL SQLDetectDeadlock(DBPROCESS *dbproc)
{
    if (*(BOOL *) dbgetuserdata(dbproc)) == TRUE)
    {
        *(BOOL *) dbgetuserdata(dbproc) = FALSE;
        return TRUE;
    }
    return FALSE;
}

/* FUNCTION: int OpenLogFile(void)
 *
 * PURPOSE:      This function opens the delivery log file for use.
 *
 * ARGUMENTS:    None
 *
 * RETURNS:      int          ERR_REGISTRY_NOT_SETUP
Registry not setup.
 *              ERR_CANNOT_CREATE_RESULTS_FILE
Cannot create results log file.
 *              ERR_SUCCESS
Log file successfully opened
 *
 * COMMENTS:     None
 *
 */

static int OpenLogFile(void)
{
    HKEY      hKey;
    BOOL      bRc;
    BYTE      szTmp[256];
    char      szKey[256];
    char      szLogPath[256];
    DWORD     size;
    DWORD     sv;

```

```

        int          len;
        char      *ptr;

        szLogPath[0] = 0;
        bRc = TRUE;
        if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\Virtual Roots", 0,
KEY_ALL_ACCESS, &hKey) == ERROR_SUCCESS )
        {
            sv = sizeof(szKey);
            size = sizeof(szTmp);

            if ( RegEnumValue(hKey, 0, szKey, &sv, NULL, NULL, szTmp, &size)
== ERROR_SUCCESS )
            {
                strcpy(szLogPath, szTmp);
                bRc = FALSE;
            }
            RegCloseKey(hKey);
        }

        if ( bRc )
            return ERR_REGISTRY_NOT_SETUP;

        if ( (ptr = strchr(szLogPath, ',')) )
            *ptr = 0;

        len = strlen(szLogPath);
        if ( szLogPath[len-1] != '\\')
        {
            szLogPath[len] = '\\';
            szLogPath[len+1] = 0;
        }
        sprintf(szLogPath, "delilog%d", GetCurrentThreadId());

        fpLog = fopen(szLogPath, "ab");

        if ( !fpLog )
            return ERR_CANNOT_CREATE_RESULTS_FILE;

        return ERR_SUCCESS;
    }

/*
 * Common Code for all Servers
 */

/* FUNCTION: BOOL SQLInit()
 *
 * PURPOSE:      This function initializes SQL Server for later use.
 *
 * RETURNS:      BOOL          FALSE          if successfull
 *              TRUE          if an error
occurs and connection cannot be established.
 *
 * COMMENTS:     None
 *
 */
BOOL SQLInit ()
{
    dbinit();

```

```

        if ( dbgetmaxprocs() < iMaxConnections )
        {
            if ( dbsetmaxprocs(iMaxConnections) == FAIL )
            {
                //set for fail error message when HttpExtensionProc()
                //at this point we don't have a pECB so no way to show
                error message.
                iMaxConnections = -1;
            }
        }

        // install error and message handlers
        dbmsghandle( (DBMSGHANDLE_PROC)msg_handler);
        dberrhandle( (DBERRHANDLE_PROC)err_handler);

        return TRUE;
    }

/* FUNCTION: BOOL SQLOpenConnection(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId, DBPROCESS **dbproc, char *server, char *database, char *user, char
*password, char *app, int *spid, long *pack_size)
*
* PURPOSE:      This function opens the sql connection for use.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*
*      iTermId      int
*                  terminal id of browser
*
*      iSyncId      int
*                  sync id of browser
*
*      DBPROCESS **dbproc pointer to
returned DBPROCESS
*
*      char *server
SQL server name
*
*      char *database SQL
server database
*
*      char *user
user name
*
*      char *password user
password
*
*      char *app
pointer to returned application array
*
*      int
*      *spid        pointer to returned spid
*
*      long
*      *pack_size  pointer to returned default pack size
*
* RETURNS:     BOOL FALSE if successfull
*              TRUE  if an error
occurs
*
* COMMENTS:    None
*/

#ifdef USE_ODBC
    static BOOL SQLOpenConnection(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
int iSyncId, DBPROCESS **dbproc, char *server, char *database, char *user, char
*password, char *app, int *spid, long *pack_size)
    {

```

```

        RETCODE rc;
        char buffer[30];

        *dbproc = (DBPROCESS *)malloc(sizeof(DBPROCESS));
        if ( !*dbproc )
            return TRUE;

        //set pECB data into dbproc
        (*dbproc)->bDeadlock = FALSE;
        (*dbproc)->bFailed = FALSE;
        (*dbproc)->pECB = pECB;
        (*dbproc)->iTermId = iTermId;
        (*dbproc)->iSyncId = iSyncId;

        if ( SQLAllocConnect(henv, &(*dbproc)->hdbc) == SQL_ERROR )
            return TRUE;

        if ( SQLSetConnectOption((*dbproc)->hdbc, SQL_PACKET_SIZE,
pack_size) == SQL_ERROR )
            return TRUE;

        rc = SQLConnect((*dbproc)->hdbc, server, SQL_NTS, user, SQL_NTS,
password, SQL_NTS);
        if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
            return TRUE;
        rc = SQLAllocStmnt((*dbproc)->hdbc, &(*dbproc)->hstmt);
        if (rc == SQL_ERROR)
            return TRUE;

        sprintf(buffer,"use %s", Client->database);

        rc = SQLExecDirect((*dbproc)->hstmt, buffer, SQL_NTS);
        if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
            return TRUE;

        SQLFreeStmnt((*dbproc)->hstmt, SQL_CLOSE);
        sprintf(buffer,"set nocount on");
        rc = SQLExecDirect((*dbproc)->hstmt, buffer, SQL_NTS);
        if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
            return TRUE;
        SQLFreeStmnt((*dbproc)->hstmt, SQL_CLOSE);

        sprintf(buffer,"select @@spid");

        rc = SQLExecDirect((*dbproc)->hstmt, buffer, SQL_NTS);
        if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
            return TRUE;

        if ( SQLBindCol((*dbproc)->hstmt, 1, SQL_C_SSHORT, &(*dbproc)-
>spid, 0, NULL) == SQL_ERROR )
            return TRUE;

        if ( SQLFetch((*dbproc)->hstmt) == SQL_ERROR )
            return TRUE;

        SQLFreeStmnt((*dbproc)->hstmt, SQL_CLOSE);

        return FALSE;
    }

#else

```

```

static BOOL SQLOpenConnection(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
int iSyncId, DBPROCESS **dbproc, char *server, char *database, char *user, char
*password, char *app, int *spid)
{
    LOGINREC *login;
    PECBINFO pEcbInfo;

    //set local msg proc for login record
    //attach pECB record

    //this is necessary as dblib provides no way to pass user data
in a login structure. So until
    //there is an allocated dbproc we need to use a static which
means that the login attempt must
    //be serialized.

    gpECB = pECB;

    login = dblogin();
    if ( !*user )
        DBSETLUSER(login, "sa");
    else
        DBSETLUSER(login, user);

    DBSETLPWD(login, password);
    DBSETLHOST(login, app);

    DBSETLPACKET(login, (unsigned short)DEFCLPACKSIZE);

    if ((*dbproc = dbopen(login, server )) == NULL)
        return TRUE;

    //set pECB data into dbproc
    pEcbInfo = (PECBINFO)malloc(sizeof(PECBINFO));
    pEcbInfo->bDeadlock = FALSE;
    pEcbInfo->pECB = pECB;
    pEcbInfo->iTermId = iTermId;
    pEcbInfo->iSyncId = iSyncId;
    dbsetuserdata(*dbproc, pEcbInfo);

    // Use the the right database
    dbuse(*dbproc, database);

    dbcmd(*dbproc, "select @@spid");

    dbsqlxec(*dbproc);
    while (dbresults(*dbproc) != NO_MORE_RESULTS)
    {
        dbbind(*dbproc, 1, SMALLBIND, (DBINT) 0, (BYTE *)
spid);
        while (dbnextrow(*dbproc) != NO_MORE_ROWS)
            ;
    }
    dbcmd(*dbproc, "set nocount on");

    dbsqlxec(*dbproc);
    while (dbresults(*dbproc) != NO_MORE_RESULTS)
    {
        while (dbnextrow(*dbproc) != NO_MORE_ROWS)
            ;
    }
}

```

```

//rollback transaction on abort
dbcmd(*dbproc, "set XACT_ABORT ON");

dbsqlxec(*dbproc);
while (dbresults(*dbproc) != NO_MORE_RESULTS)
{
    while (dbnextrow(*dbproc) != NO_MORE_ROWS)
        ;
}

return FALSE;
}

#endif

/* FUNCTION: BOOL SQLCloseConnection(EXTENSION_CONTROL_BLOCK *pECB, DBPROCESS
*dbproc)
*
* PURPOSE: This function closes the sql connection.
*
* ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
* DBPROCESS *dbproc pointer to
DBPROCESS
*
* RETURNS: BOOL FALSE if successfull
TRUE if an error
occurs
*
* COMMENTS: None
*/

#ifdef USE_ODBC
static BOOL SQLCloseConnection(EXTENSION_CONTROL_BLOCK *pECB, DBPROCESS
*dbproc)
{
    if ( dbproc )
    {
        SQLFreeStmt(dbproc->hstmt, SQL_DROP);
        SQLDisconnect(dbproc->hdbc);
        SQLFreeConnect(dbproc->hdbc);
        free(dbproc);
        dbproc = NULL;
    }
    return FALSE;
}
#else
static BOOL SQLCloseConnection(EXTENSION_CONTROL_BLOCK *pECB, DBPROCESS
*dbproc)
{
    if (dbclose(dbproc) == FAIL)
        return TRUE;
    return FALSE;
}
#endif

// Lifted from HP FDR since they did such a nice job
void TMLog( char *format, ... )
{
    va_list args;
    char buf[4096];
    int len;
}

```

```

        va_start( args, format );
        _strtime( buf );
        strcat( buf, " ");
        len = strlen( buf );
        (void)_vsnprintf( buf+ len, sizeof( buf) - len - 1, format, args);
        buf[sizeof( buf) - 1] = '\0';
        va_end( args );
        userlog( buf );
    }

/* FUNCTION: void UtilStrCpy(char *pDest, char *pSrc, int n)
 *
 * PURPOSE:      This function copies n characters from string pSrc to pDst and
places a
 *
 *               null character at the end of the destination string.
 *
 * ARGUMENTS:   char          *pDest  destination string
pointer
 *
 *               source string pointer      char          *pSrc
 *
 *               number of characters to copy      int          n
 *
 * RETURNS:     None
 *
 * COMMENTS:    Unlike strncpy this function ensures that the result string is
 *               always null terminated.
 */

static void UtilStrCpy(char *pDest, char *pSrc, int n)
{
    strncpy(pDest, pSrc, n);
    pDest[n] = '\0';

    return;
}

/* FUNCTION: int err_handler(DBPROCESS *dbproc, int severity, int dberr, int oserr,
char *dberrstr, char *oserrstr)
 *
 * PURPOSE:      This function handles DB-Library errors
 *
 * ARGUMENTS:   DBPROCESS      *dbproc      DBPROCESS id
pointer
 *
 *               severity      severity of error      int
 *
 *               dberr         error id      int
 *
 *               oserr         operating system specific error code      int
 *
 *               printable error description of dberr      char          *dberrstr
 *
 *               printable error description of oserr      char          *oserrstr
 *
 * RETURNS:     int          INT_CONTINUE
 *               continue if error is SQLETIME else INT_CANCEL action
 *
 * COMMENTS:    None
 */

```

```

int err_handler(DBPROCESS *dbproc, int severity, int dberr, int oserr, char
 *dberrstr, char *oserrstr)
{
    PECBINFO          pEcbInfo;
    EXTENSION_CONTROL_BLOCK *pECB;
    FILE              *fp;
    SYSTEMTIME        systemTime;
    char              szTmp[256];
    int               iTermId;
    int               iSyncId;

    pEcbInfo = NULL;

    if ((dbproc == NULL) || (DBDEAD(dbproc)))
    {
        TMLog("DBPROC is invalid");
        return INT_CANCEL;
    }

    if ( !(pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
    {
        pECB = gpECB;
        iTermId = 0;
        iSyncId = 0;
    }
    else
    {
        pECB = pEcbInfo->pECB;
        iTermId = pEcbInfo->iTermId;
        iSyncId = pEcbInfo->iSyncId;
    }

    if ( pEcbInfo && pEcbInfo->bFailed )
    {
        bError == FALSE;
        return INT_CANCEL;
    }

    if ( oserr != DENOERR )
    {
        TMLog("DBLIB Error %s", oserrstr);
        if ( pEcbInfo )
        {
            pEcbInfo->bFailed = TRUE;
            bError = TRUE;
        }

        GetLocalTime(&systemTime);
        fp = fopen(szErrorLogPath, "ab");

        sprintf(szTmp, "ErrorHandler: DBLIB(%d): %s", oserr, oserrstr);

        TMLog("%2.2d/%2.2d/%2.2d %2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",
            systemTime.wYear, systemTime.wMonth, systemTime.wDay,
            systemTime.wHour, systemTime.wMinute,
            szTmp);

        fclose(fp);
    }

    return INT_CANCEL;
}

```

```

}

/* FUNCTION: int msg_handler(DBPROCESS *dbproc, DBINT msgno, int msgstate, int
severity, char *msgtext)
*
* PURPOSE:      This function handles DB-Library SQL Server error messages
*
* ARGUMENTS:   DBPROCESS      *dbproc      DBPROCESS id
pointer
*
*              message number      DBINT      msgno
*
*              message state      int
* msgstate      message state      int
*
*              message severity    char
* severity      message severity    *msgtext
*
*              printable message description
*
* RETURNS:     int      INT_CONTINUE
*              continue if error is SQLETIME else INT_CANCEL action
*
*              INT_CANCEL      cancel operation
*
* COMMENTS:    This function also sets the dead lock dbproc variable if
necessary.
*
*/

int msg_handler(DBPROCESS *dbproc, DBINT msgno, int msgstate, int severity, char
*msgtext)
{
    PECBINFO
    EXTENSION_CONTROL_BLOCK      *pECB;      pEcbInfo;
    FILE
    SYSTEMTIME                    *fp;
    char                          systemTime;
    int                           szTmp[256];
    int                           iTermId;
    int                           iSyncId;

    if ( !(pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
    {
        pECB = gpECB;
        iTermId = 0;
        iSyncId = 0;
    }
    else
    {
        pECB = pEcbInfo->pECB;
        iTermId = pEcbInfo->iTermId;
        iSyncId = pEcbInfo->iSyncId;
    }

    if ( (msgno == 5701) || (msgno == 2528) || (msgno == 5703) || (msgno ==
6006) )
        return INT_CONTINUE;

    // deadlock message
    if (msgno == 1205)
    {
        // set the deadlock indicator
        if ( pEcbInfo )

```

```

        pEcbInfo->bDeadlock = TRUE;
    else
        TMLog("Error, dbgetuserdata returned NULL.");
    return INT_CONTINUE;
}
if ( pEcbInfo && pEcbInfo->bFailed )
{
    TMLog("SQL Error ");
    return INT_CANCEL;
}
if (msgno == 0)
    return INT_CONTINUE;
else
{
    TMLog("MsgHandler: SQL Error %s", msgtext);

    if ( pEcbInfo )
        pEcbInfo->bFailed = TRUE;

    bError = TRUE;

    sprintf(szTmp, "Error: SQLSVR(%d): %s", msgno, msgtext);

    TMLog("%2.2d/%2.2d/%2.2d %2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",
systemTime.wYear, systemTime.wMonth, systemTime.wDay,
systemTime.wHour, systemTime.wMinute,
systemTime.wSecond,
szTmp);
}
return INT_CANCEL;
}

/* FUNCTION: BOOL GetParameters(int argc, char *argv[])
*
* PURPOSE:      This function parses the command line passed in to the delivery
executable, initializing
*
*              and filling in global variable parameters.
*
* ARGUMENTS:   int      argc      number of command line arguments
passed to delivery
*
*              char      *argv[]  array of command line
argument pointers
*
* RETURNS:     BOOL      FALSE    parameter read successfull
*              TRUE      user has
requested parameter information screen be displayed.
*
* COMMENTS:    None
*
*/

static BOOL GetParameters(int argc, char *argv[])
{
    int i;

    szServer[0]          = 0;
    szPassword[0]       = 0;
    bFlush               = FALSE;
    strcpy(szDatabase, "tpcc");
    strcpy(szUser, "sa");

```

```

for(i=0; i<argc; i++)
{
    if ( argv[i][0] == '-' || argv[i][0] == '/' )
    {
        switch(argv[i][1])
        {
            case 'S':
            case 's':
                strcpy(szServer, argv[i+2]);
                break;
            case 'V':
            case 'v':
                verbose = TRUE;
                break;
            case 'F':
            case 'f':
                bFlush = TRUE;        //turn on
delilog flush when written.
                break;
            case '?':
                return TRUE;
        }
    }
}
return FALSE;
}

/* FUNCTION: void PrintParameters(void)
*
* PURPOSE:      This function displays the supported command line flags.
*
* ARGUMENTS:   None
*
* RETURNS:     None
*
* COMMENTS:    None
*/

static void PrintParameters(void)
{
    TMLog("Performance Tuning Corporation Tuxedo Kit");
    TMLog(" www.perftuning.com (281) 251-3495 ");
    TMLog("Delivery: -S Server -D Database -U User -P Password -F ");
    TMLog("Delivery: Server %s, Database %s, User %s, Password %s, Flush %d.",
        szServer, szDatabase, szUser, szPassword, bFlush);
}

```

getval.h

```

/* MACRO: GetVal(char *pQueryString, char *pKey, char *pValue, int iMax)
*
* PURPOSE:      This function parses a http formatted string for specific key
values.
*
* ARGUMENTS:   char          *pQueryString    http string
from client browser
               char          *pKey
               key value to look for

```

```

*
* char          *pValue
character array into which to place key's value
*
* int          iMax
maximum length of key value array.
*
* RETURNS:     BOOL          FALSE    key value not found
*
*              TRUE         key valud
found
*
*
* COMMENTS:    http keys are formatted either KEY=value& or KEY=value\0. This
DLL formats
*
*              TPC-C input fields in such a manner that the
keys can be extracted in the
*
*              above manner.
*/

#define GetVal(s, k, v, m) \
{ int max = m; char * vp = v; \
  if ( (s=strstr(s, k)) ) \
    if ( (s=strchr(s, '=')) ) \
    { \
      s++; \
      max--; \
      while( *s && *s != '&' && max) \
      { \
        *vp++ = *s++; \
        max--; \
      } \
      *vp = 0; \
      s-- ; \
    } \
}

```

httpext.h

```

/*****
*
* Copyright (c) 1995 Process Software Corporation
*
* Copyright (c) 1995 Microsoft Corporation
*
*
* Module Name : HttpExt.h
*
* Abstract :
*
* This module contains the structure definitions and prototypes for the
* version 1.0 HTTP Server Extension interface.
*
*****/

#ifndef _HTTPEXT_H_
#define _HTTPEXT_H_

#include <windows.h>

#ifdef __cplusplus
extern "C" {
#endif

#define HSE_VERSION_MAJOR 1 // major version of this spec

```

```

#define HSE_VERSION_MINOR          0      // minor version of this spec
#define HSE_LOG_BUFFER_LEN        80
#define HSE_MAX_EXT_DLL_NAME_LEN  256

typedef LPVOID HCONN;

// the following are the status codes returned by the Extension DLL

#define HSE_STATUS_SUCCESS          1
#define HSE_STATUS_SUCCESS_AND_KEEP_CONN  2
#define HSE_STATUS_PENDING         3
#define HSE_STATUS_ERROR           4

// The following are the values to request services with the ServerSupportFunction.
// Values from 0 to 1000 are reserved for future versions of the interface

#define HSE_REQ_BASE                0
#define HSE_REQ_SEND_URL_REDIRECT_RESP ( HSE_REQ_BASE + 1 )
#define HSE_REQ_SEND_URL            ( HSE_REQ_BASE + 2 )
#define HSE_REQ_SEND_RESPONSE_HEADER ( HSE_REQ_BASE + 3 )
#define HSE_REQ_DONE_WITH_SESSION   ( HSE_REQ_BASE + 4 )
#define HSE_REQ_END_RESERVED        1000

//
// These are Microsoft specific extensions
//

#define HSE_REQ_MAP_URL_TO_PATH      (HSE_REQ_END_RESERVED+1)
#define HSE_REQ_GET_SSPI_INFO       (HSE_REQ_END_RESERVED+2)

//
// passed to GetExtensionVersion
//

typedef struct _HSE_VERSION_INFO {
    DWORD dwExtensionVersion;
    CHAR lpszExtensionDesc[HSE_MAX_EXT_DLL_NAME_LEN];
} HSE_VERSION_INFO, *LPHSE_VERSION_INFO;

//
// passed to extension procedure on a new request
//

typedef struct _EXTENSION_CONTROL_BLOCK {
    DWORD cbSize; // size of this struct.
    DWORD dwVersion; // version info of this spec
    HCONN ConnID; // Context number not to be modified!
    DWORD dwHttpStatusCode; // HTTP Status code
    CHAR lpszLogData[HSE_LOG_BUFFER_LEN]; // null terminated log info specific
to this Extension DLL

    LPSTR lpszMethod; // REQUEST_METHOD
    LPSTR lpszQueryString; // QUERY_STRING
    LPSTR lpszPathInfo; // PATH_INFO
    LPSTR lpszPathTranslated; // PATH_TRANSLATED

    DWORD cbTotalBytes; // Total bytes indicated from client
    DWORD cbAvailable; // Available number of bytes
    LPBYTE lpbData; // pointer to cbAvailable bytes

```

```

LPSTR lpszContentType; // Content type of client data

BOOL (WINAPI * GetServerVariable) ( HCONN hConn,
LPSTR lpszVariableName,

LPVOID lpvBuffer,
LPDWORD lpdwSize );

BOOL (WINAPI * WriteClient) ( HCONN ConnID,
LPVOID Buffer,
LPDWORD lpdwBytes,
DWORD dwReserved );

BOOL (WINAPI * ReadClient) ( HCONN ConnID,
LPVOID lpvBuffer,
LPDWORD lpdwSize );

BOOL (WINAPI * ServerSupportFunction) ( HCONN hConn,
DWORD dwHSERequest,
LPVOID lpvBuffer,
LPDWORD lpdwSize,
LPDWORD lpdwDataType );

} EXTENSION_CONTROL_BLOCK, *LPEXTENSION_CONTROL_BLOCK;

//
// these are the prototypes that must be exported from the extension DLL
//

BOOL WINAPI GetExtensionVersion( HSE_VERSION_INFO *pVer );
DWORD WINAPI HttpExtensionProc( EXTENSION_CONTROL_BLOCK *pECB );

// the following type declarations is for the server side

typedef BOOL (WINAPI * PFN_GETEXTENSIONVERSION) ( HSE_VERSION_INFO *pVer );
typedef DWORD (WINAPI * PFN_HTTPEXTENSIONPROC) ( EXTENSION_CONTROL_BLOCK *pECB );

#ifdef __cplusplus
}
#endif

#endif // end definition _HTTPEXT_H_

```

install.c

```

/* FILE: INSTALL.C
 * Microsoft TPC-C Kit Ver. 3.00.000
 * Audited 08/23/96, By Francois Raab
 *
 * Copyright Microsoft, 1996
 *
 * PURPOSE: Automated installation application for TPC-C Web Kit
 * Author: Philip Durr
 * philipdu@Microsoft.com
 */

#include <windows.h>
#include <direct.h>

```

```

#include <io.h>
#include <stdlib.h>
#include <stdio.h>
#include <commctrl.h>
#include "install.h"

HICON          hIcon;
HINSTANCE hInst;

DWORD          versionExeMS;
DWORD          versionExeLS;
DWORD          versionDllMS;
DWORD          versionDllLS;

static BOOL    bLog;
static int     iThreads;
static int     iMaxWarehouse;
static int     iDelayMs;
static int     iDeadlockRetry;
static int     iMaxConnections;
static int     iPoolThreadsLimit;
static int     iThreadTimeout;
static int     iListenBackLog;
static int     iAcceptExOutstanding;
static int     iQSlotts;

static int     iMaxPhysicalMemory; //max physical memory in
MB

BOOL CALLBACK UpdatedDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM
lParam);
BOOL CALLBACK MainDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM lParam);
BOOL CALLBACK CopyDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM lParam);
static void    ReadRegistrySettings(void);
static void    WriteRegistrySettings(char *szDllPath);
static int     CopyFiles(HWND hDlg, char *szDllPath);
static BOOL    GetInstallPath(char *szDllPath);
static void    GetVersionInfo(char *szDLLPath, char *szExePath);
static BOOL    StartWWWebService(void);
static BOOL    StopWWWebService(void);
static void    UpdateDialog(HWND hDlg);

int WINAPI WinMain( HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpCmdLine,
int nCmdShow )
{
    int iRc;

    hInst = hInstance;

    InitCommonControls();

    hIcon = LoadIcon(hInstance, MAKEINTRESOURCE(IDI_ICON1));

    iRc = DialogBox(hInstance, MAKEINTRESOURCE(IDD_DIALOG1),
GetDesktopWindow(), MainDlgProc);
    if ( iRc )
        DialogBoxParam(hInstance,
MAKEINTRESOURCE(IDD_DIALOG2), GetDesktopWindow(), UpdatedDlgProc, (LPARAM) iRc);
    DestroyIcon(hIcon);

    return 0;
}

```

```

BOOL CALLBACK UpdatedDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM lParam)
{
    switch(uMsg)
    {
        case WM_INITDIALOG:
            if ( lParam == 1 )
                SetDlgItemText(hwnd, IDC_RESULTS, "HTML TPCC
Installation Successful");
            else
                SetDlgItemText(hwnd, IDC_RESULTS, "HTML TPCC
Registry Updated");
            return TRUE;
        case WM_COMMAND:
            if ( wParam == IDOK )
                EndDialog(hwnd, TRUE);
            break;
        default:
            break;
    }
    return FALSE;
}

BOOL CALLBACK MainDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM lParam)
{
    PAINTSTRUCT ps;
    MEMORYSTATUS memoryStatus;
    int          d;
    int          rc;
    HWND         hDlg;
    char         szTmp[256];
    static char  szDllPath[256];
    static char  szExePath[256];

    switch(uMsg)
    {
        case WM_INITDIALOG:
            GlobalMemoryStatus(&memoryStatus);
            iMaxPhysicalMemory = (memoryStatus.dwTotalPhys/
1048576);

            if ( GetInstallPath(szDllPath) )
            {
                MessageBox(hwnd, "Error internet service
inetsrv is not installed.", NULL, MB_ICONSTOP | MB_OK);
                EndDialog(hwnd, FALSE);
                return TRUE;
            }

            bLog = FALSE;
            iThreads = 4;
            iMaxWarehouse = 0;
            iDelayMs = 500;
            iDeadlockRetry = 3;
            iMaxConnections = 25;
            iPoolThreadsLimit = iMaxPhysicalMemory *
2;
            iThreadTimeout = 86400;
            iListenBackLog = 15;
            iAcceptExOutstanding = 40;

            ReadRegistrySettings();
    }
}

```



```

sizeof(szExePath));
    GetModuleFileName(hInst, szExePath,
    GetVersionInfo(szDllPath, szExePath);
    if ( bLog )
        CheckDlgButton(hwnd, BN_LOG, 1);

    wsprintf(szTmp, "Version %d.00.%3.3d", versionExeMS,
versionExeLS);

    SetDlgItemText(hwnd, IDC_VERSION, szTmp);

    SetDlgItemText(hwnd, IDC_PATH, szDllPath);
    SetDlgItemInt(hwnd, ED_MAXWARE, iMaxWareHouse, FALSE);
    SetDlgItemInt(hwnd, ED_THREADS, iThreads, FALSE);
    SetDlgItemInt(hwnd, ED_MAXCONNECTION, iMaxConnections,
FALSE);
    SetDlgItemInt(hwnd, ED_IIS_MAX_THREAD_POOL_LIMIT,
iPoolThreadsLimit, FALSE);
    SetDlgItemInt(hwnd, ED_IIS_THREAD_TIMEOUT,
iThreadTimeout, FALSE);
    SetDlgItemInt(hwnd, ED_IIS_LISTEN_BACKLOG,
iListenBackLog, FALSE);
    SetDlgItemInt(hwnd, ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE,
iAcceptExOutstanding, FALSE);

    return TRUE;
case WM_PAINT:
    if ( IsIconic(hwnd) )
    {
        BeginPaint(hwnd, &ps);
        DrawIcon(ps.hdc, 0, 0, hIcon);
        EndPaint(hwnd, &ps);
        return TRUE;
    }
    break;
case WM_COMMAND:
    if ( wParam == IDOK )
    {
        if ( IsDlgButtonChecked(hwnd, BN_LOG) )
            bLog = TRUE;
        else
            bLog = FALSE;
        iThreads = GetDlgItemInt(hwnd, ED_THREADS,
&d, FALSE);
        iMaxWareHouse = GetDlgItemInt(hwnd,
ED_MAXWARE, &d, FALSE);
        iMaxConnections = GetDlgItemInt(hwnd,
ED_MAXCONNECTION, &d, FALSE);

        iPoolThreadsLimit = GetDlgItemInt(hwnd,
ED_IIS_MAX_THREAD_POOL_LIMIT, &d, FALSE);
        iThreadTimeout = GetDlgItemInt(hwnd,
ED_IIS_THREAD_TIMEOUT, &d, FALSE);
        iListenBackLog = GetDlgItemInt(hwnd,
ED_IIS_LISTEN_BACKLOG, &d, FALSE);
        iAcceptExOutstanding = GetDlgItemInt(hwnd,
ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE, &d, FALSE);

        ShowWindow(hwnd, SW_HIDE);
        hDlg = CreateDialog(hInst,
MAKEINTRESOURCE(IDD_DIALOG3), hwnd, CopyDlgProc);
        ShowWindow(hDlg, SW_SHOWNA);
        UpdateDialog(hDlg);

```

```

rc = CopyFiles(hDlg, szDllPath);
if ( !rc )
{
    ShowWindow(hwnd, SW_SHOWNA);
    DestroyWindow(hDlg);
    MessageBox(hwnd, "Error(s) occurred
when creating tpcc.dll", NULL, MB_ICONSTOP | MB_OK);
    EndDialog(hwnd, 0);
    return TRUE;
}
SetDlgItemText(hDlg, IDC_STATUS, "Updating
Registry.");
SendDlgItemMessage(hDlg, IDC_PROGRESS1,
PBM_STEPIT, 0, 0);

UpdateDialog(hDlg);
WriteRegistrySettings(szDllPath);

Sleep(100);

ShowWindow(hwnd, SW_SHOWNA);
DestroyWindow(hDlg);

EndDialog(hwnd, rc);
return TRUE;
}
if ( wParam == IDCANCEL )
{
    EndDialog(hwnd, FALSE);
    return TRUE;
}
break;
default:
    break;
}
return FALSE;
}

static void ReadRegistrySettings(void)
{
    HKEY    hKey;
    DWORD  size;
    DWORD  type;
    char    szTmp[256];

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE, "SOFTWARE\\Microsoft\\TPCC", 0,
KEY_READ, &hKey) == ERROR_SUCCESS )
    {
        size = sizeof(szTmp);
        bLog = FALSE;
        if ( RegQueryValueEx(hKey, "LOG", 0, &type, szTmp, &size) ==
ERROR_SUCCESS )
            if ( !strcmp(szTmp, "ON") )
                bLog = TRUE;

        iThreads = 4;
        size = sizeof(szTmp);
        if ( RegQueryValueEx(hKey, "NumberOfDeliveryThreads", 0, &type,
szTmp, &size) == ERROR_SUCCESS )
            iThreads = atoi(szTmp);
        if ( iThreads == 0 )
            iThreads = 4;

```

```

        iMaxWareHouse = 0;
        size = sizeof(szTmp);
        if ( RegQueryValueEx(hKey, "MaximumWarehouses", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
            iMaxWareHouse = atoi(szTmp);
        if ( iMaxWareHouse == 0 )
            iMaxWareHouse = 0;

        iDelayMs = 500;
        size = sizeof(szTmp);
        if ( RegQueryValueEx(hKey, "BackoffDelay", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
            iDelayMs = atoi(szTmp);
        if ( iDelayMs == 0 )
            iDelayMs = 500;

        iDeadlockRetry = 3;
        size = sizeof(szTmp);
        if ( RegQueryValueEx(hKey, "DeadlockRetry", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
            iDeadlockRetry = atoi(szTmp);
        if ( !iDeadlockRetry )
            iDeadlockRetry = 3;

        iMaxConnections = 25;
        size = sizeof(szTmp);
        if ( RegQueryValueEx(hKey, "MaxConnections", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
            iMaxConnections = atoi(szTmp);
        if ( !iMaxConnections )
            iMaxConnections = 25;

        iQSlotts = 3000;
        size = sizeof(szTmp);
        if ( RegQueryValueEx(hKey, "QueueSlotts", 0, &type, szTmp,
&size) == ERROR_SUCCESS )
            iQSlotts = atoi(szTmp);
        if ( iQSlotts == 0 )
            iQSlotts = 3000;

        RegCloseKey(hKey);

        if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\Inetinfo\\Parameters", 0, KEY_READ, &hKey) ==
ERROR_SUCCESS )
        {
            iPoolThreadsLimit = iMaxPhysicalMemory * 2;
            size = sizeof(iPoolThreadsLimit);
            if ( RegQueryValueEx(hKey, "PoolThreadsLimit", 0,
&type, (char *)&iPoolThreadsLimit, &size) == ERROR_SUCCESS )
                if ( !iPoolThreadsLimit )
                    iPoolThreadsLimit = iMaxPhysicalMemory * 2;

            iThreadTimeout = 86400;
            size = sizeof(iThreadTimeout);
            if ( RegQueryValueEx(hKey, "ThreadTimeout", 0, &type,
(char *)&iThreadTimeout, &size) == ERROR_SUCCESS )
                if ( !iThreadTimeout )
                    iThreadTimeout = 86400;

            iListenBackLog = 15;
            size = sizeof(iListenBackLog);

```

```

        if ( RegQueryValueEx(hKey, "ListenBackLog", 0, &type,
(char *)&iListenBackLog, &size) == ERROR_SUCCESS )
            if ( !iListenBackLog )
                iListenBackLog = 15;
        }

        RegCloseKey(hKey);

        if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters", 0, KEY_READ, &hKey) ==
ERROR_SUCCESS )
        {
            iAcceptExOutstanding = 40;
            size = sizeof(iAcceptExOutstanding);
            if ( RegQueryValueEx(hKey, "AcceptExOutstanding", 0,
&type, (char *)&iAcceptExOutstanding, &size) == ERROR_SUCCESS )
                if ( !iAcceptExOutstanding )
                    iAcceptExOutstanding = 40;
        }

        RegCloseKey(hKey);
    }

    return;
}

static void WriteRegistrySettings(char *szDllPath)
{
    HKEY    hKey;
    DWORD   dwDisposition;
    char    szTmp[256];
    char    *ptr;
    int     iRc;

    if ( RegCreateKeyEx(HKEY_LOCAL_MACHINE, "SOFTWARE\\Microsoft\\TPCC", 0,
NULL, REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS, NULL, &hKey, &dwDisposition) ==
ERROR_SUCCESS )
    {
        strcpy(szTmp, szDllPath);
        ptr = strstr(szTmp, "tpcc");
        if ( ptr )
            *ptr = 0;

        RegSetValueEx(hKey, "PATH", 0, REG_SZ, szTmp, strlen(szTmp));

        if ( bLog )
            RegSetValueEx(hKey, "LOG", 0, REG_SZ, "ON", 2);
        else
            RegSetValueEx(hKey, "LOG", 0, REG_SZ, "OFF", 3);

        itoa(iThreads, szTmp, 10);
        RegSetValueEx(hKey, "NumberOfDeliveryThreads", 0, REG_SZ, szTmp,
strlen(szTmp));

        itoa(iMaxWareHouse, szTmp, 10);
        RegSetValueEx(hKey, "MaximumWarehouses", 0, REG_SZ, szTmp,
strlen(szTmp));

        itoa(iDelayMs, szTmp, 10);
        RegSetValueEx(hKey, "BackoffDelay", 0, REG_SZ, szTmp,
strlen(szTmp));

        itoa(iDeadlockRetry, szTmp, 10);

```

```

        RegSetValueEx(hKey, "DeadlockRetry", 0, REG_SZ, szTmp,
strlen(szTmp));

        itoa(iMaxConnections, szTmp, 10);
        RegSetValueEx(hKey, "MaxConnections", 0, REG_SZ, szTmp,
strlen(szTmp));

        itoa(iQSlotts, szTmp, 10);
        RegSetValueEx(hKey, "QueueSlotts", 0, REG_SZ, szTmp,
strlen(szTmp));

        RegFlushKey(hKey);

        RegCloseKey(hKey);
    }

    if ( (iRc=RegCreateKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\Inetinfo\\Parameters", 0, NULL,
REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS, NULL, &hKey, &dwDisposition)) ==
ERROR_SUCCESS )
    {
        RegSetValueEx(hKey, "PoolThreadsLimit", 0, REG_DWORD, (char
*)&iPoolThreadsLimit, sizeof(iPoolThreadsLimit));
        RegSetValueEx(hKey, "ThreadTimeout", 0, REG_DWORD, (char
*)&iThreadTimeout, sizeof(iThreadTimeout));
        RegSetValueEx(hKey, "ListenBackLog", 0, REG_DWORD, (char
*)&iListenBackLog, sizeof(iListenBackLog));

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

    if ( (iRc=RegCreateKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters", 0, NULL,
REG_OPTION_NON_VOLATILE, KEY_ALL_ACCESS, NULL, &hKey, &dwDisposition)) ==
ERROR_SUCCESS )
    {
        RegSetValueEx(hKey, "AcceptExOutstanding", 0, REG_DWORD, (char
*)&iAcceptExOutstanding, sizeof(iAcceptExOutstanding));

        RegFlushKey(hKey);
        RegCloseKey(hKey);
    }

    return;
}

BOOL CALLBACK CopyDlgProc(HWND hwnd, UINT uMsg, WPARAM wParam, LPARAM lParam)
{
    if ( uMsg == WM_INITDIALOG )
    {
        SendDlgItemMessage(hwnd, IDC_PROGRESS1, PBM_SETRANGE, 0,
MAKELPARAM(0, 8));
        SendDlgItemMessage(hwnd, IDC_PROGRESS1, PBM_SETSTEP, (WPARAM)1,
0);

        return TRUE;
    }
    return FALSE;
}

static int CopyFiles(HWND hDlg, char *szDllPath)
{
    HGLOBAL          hDLL;

```

```

    HGLOBAL          hExe;
    HRSRC            hResInfo;
    BYTE            *pSrc;
    HANDLE          hFile;
    DWORD           dwSize;
    DWORD           d;
    char            szTmp[256];
    char            *ptr;
    BOOL            bSvcRunning;

    SetDlgItemText(hDlg, IDC_STATUS, "Stopping Web Service.");
    SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);

    bSvcRunning = !StopWWWService();
    SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);

    hResInfo = FindResource(hInst, MAKEINTRESOURCE(IDR_TPCCDLL), "TPCCDLL");
    SetDlgItemText(hDlg, IDC_STATUS, "Copying Files...");
    SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);

    dwSize = SizeofResource(hInst, hResInfo);
    hDLL = LoadResource(hInst, hResInfo);
    pSrc = (BYTE *)LockResource(hDLL);
    remove(szDllPath);

    if ( !(hFile = CreateFile(szDllPath, GENERIC_WRITE, 0, NULL,
CREATE_ALWAYS, FILE_ATTRIBUTE_NORMAL, NULL)) )
        return 0;

    if ( !WriteFile(hFile, pSrc, dwSize, &d, NULL) )
        return 0;

    CloseHandle(hFile);

    UnlockResource(hDLL);
    FreeResource(hDLL);

    SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);

    hResInfo = FindResource(hInst, MAKEINTRESOURCE(IDR_DELIVERY1),
"DELIVERY");

    dwSize = SizeofResource(hInst, hResInfo);
    hExe = LoadResource(hInst, hResInfo);
    pSrc = (BYTE *)LockResource(hExe);

    strcpy(szTmp, szDllPath);
    ptr = strstr(szTmp, "tpcc");
    if ( ptr )
        *ptr = 0;
    strcat(szTmp, "delisrv.exe");

    remove(szTmp);

    if ( !(hFile = CreateFile(szTmp, GENERIC_WRITE, 0, NULL, CREATE_ALWAYS,
FILE_ATTRIBUTE_NORMAL, NULL)) )
        return 0;

    if ( !WriteFile(hFile, pSrc, dwSize, &d, NULL) )

```

```

        return 0;

    CloseHandle(hFile);

    UnlockResource(hExe);
    FreeResource(hExe);

    SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);

    //if we stopped service restart it.
    if ( !bSvcRunning )
    {
        SetDlgItemText(hDlg, IDC_STATUS, "Starting Web Service.");
        SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0, 0);
        UpdateDialog(hDlg);
        StartWWWWebService();
    }

    SendDlgItemMessage(hDlg, IDC_PROGRESS1, PBM_STEPIT, 0, 0);
    UpdateDialog(hDlg);

    return 1;
}

static BOOL GetInstallPath(char *szDllPath)
{
    HKEY hKey;
    BYTE szTmp[256];
    char szKey[256];
    DWORD size;
    DWORD sv;
    BOOL bRc;
    int len;
    char *ptr;

    szDllPath[0] = 0;
    bRc = TRUE;
    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SYSTEM\\CurrentControlSet\\Services\\W3SVC\\Parameters\\Virtual Roots", 0,
KEY_ALL_ACCESS, &hKey) == ERROR_SUCCESS )
    {
        sv = sizeof(szKey);
        size = sizeof(szTmp);
        if ( RegEnumValue(hKey, 0, szKey, &sv, NULL, NULL, szTmp, &size)
== ERROR_SUCCESS )
        {
            strcpy(szDllPath, szTmp);
            bRc = FALSE;
        }
        RegCloseKey(hKey);
    }
    if ( (ptr = strchr(szDllPath, ',')) )
        *ptr = 0;

    len = strlen(szDllPath);
    if ( szDllPath[len-1] != '\\')
    {
        szDllPath[len] = '\\';
        szDllPath[len+1] = 0;
    }
    strcat(szDllPath, "tpcc.dll");
}

```

```

        return bRc;
    }

    static void GetVersionInfo(char *szDLLPath, char *szExePath)
    {
        DWORD d;
        DWORD dwSize;
        DWORD dwBytes;
        char *ptr;
        VS_FIXEDFILEINFO *vs;

        versionDllMS = 0;
        versionDllLS = 0;
        if ( _access(szDLLPath, 00) == 0 )
        {
            dwSize = GetFileVersionInfoSize(szDLLPath, &d);
            if ( dwSize )
            {
                ptr = (char *)malloc(dwSize);
                GetFileVersionInfo(szDLLPath, 0, dwSize, ptr);
                VerQueryValue(ptr, "\\",&vs, &dwBytes);
                versionDllMS = vs->dwProductVersionMS;
                versionDllLS = vs->dwProductVersionLS;
                free(ptr);
            }
        }

        versionExeMS = 0x7FFF;
        versionExeLS = 0x7FFF;
        dwSize = GetFileVersionInfoSize(szExePath, &d);
        if ( dwSize )
        {
            ptr = (char *)malloc(dwSize);
            GetFileVersionInfo(szExePath, 0, dwSize, ptr);
            VerQueryValue(ptr, "\\",&vs, &dwBytes);

            versionExeMS = vs->dwProductVersionMS;
            versionExeLS = vs->dwProductVersionLS;
            free(ptr);
        }
        return;
    }

    static BOOL StartWWWWebService(void)
    {
        SC_HANDLE schSCManager;
        SC_HANDLE schService;
        SERVICE_STATUS ssStatus;
        DWORD dwOldCheckPoint;

        schSCManager = OpenSCManager(NULL, NULL, SC_MANAGER_ALL_ACCESS);
        schService = OpenService(schSCManager, TEXT("W3SVC"), SERVICE_ALL_ACCESS);
        if (schService == NULL)
            return FALSE;

        if (! StartService(schService, 0, NULL) )
            goto StartWWWWebErr;
        //start Service pending, Check the status until the service is running.
        if (! QueryServiceStatus(schService, &ssStatus) )
            goto StartWWWWebErr;
        while( ssStatus.dwCurrentState != SERVICE_RUNNING)
        {

```

```

        dwOldCheckPoint = ssStatus.dwCheckPoint;
//Save the current checkpoint.
        Sleep(ssStatus.dwWaitHint);
        //Wait for the specified interval.
        if ( !QueryServiceStatus(schService, &ssStatus) ) //Check the
status again.
            break;
        if (dwOldCheckPoint >= ssStatus.dwCheckPoint)
//Break if the checkpoint has not been incremented.
            break;
    }

    if (ssStatus.dwCurrentState == SERVICE_RUNNING)
        goto StartWWWebErr;

    CloseServiceHandle(schService);
    return TRUE;

StartWWWebErr:
    CloseServiceHandle(schService);
    return FALSE;
}

static BOOL StopWWWebService(void)
{
    SC_HANDLE          schSCManager;
    SC_HANDLE          schService;
    SERVICE_STATUS     ssStatus;
    DWORD              dwOldCheckPoint;

    schSCManager = OpenSCManager(NULL, NULL, SC_MANAGER_ALL_ACCESS);
    schService = OpenService(schSCManager, TEXT("W3SVC"), SERVICE_ALL_ACCESS);
    if (schService == NULL)
        return FALSE;

    if ( ! QueryServiceStatus(schService, &ssStatus) )
        goto StopWWWebErr;

    if ( !ControlService(schService, SERVICE_CONTROL_STOP, &ssStatus) )
        goto StopWWWebErr;
//start Service pending, Check the status until the service is running.
    if ( ! QueryServiceStatus(schService, &ssStatus) )
        goto StopWWWebErr;
    while( ssStatus.dwCurrentState == SERVICE_RUNNING)
    {
        dwOldCheckPoint = ssStatus.dwCheckPoint;
//Save the current checkpoint.
        Sleep(ssStatus.dwWaitHint);
        //Wait for the specified interval.
        if ( !QueryServiceStatus(schService, &ssStatus) ) //Check the
status again.
            break;
        if (dwOldCheckPoint >= ssStatus.dwCheckPoint)
//Break if the checkpoint has not been incremented.
            break;
    }

    if (ssStatus.dwCurrentState == SERVICE_RUNNING)
        goto StopWWWebErr;

    CloseServiceHandle(schService);

```

```

        return TRUE;

StopWWWebErr:
    CloseServiceHandle(schService);
    return FALSE;
}

static void UpdateDialog(HWND hDlg)
{
    MSG msg;

    UpdateWindow(hDlg);
    while( PeekMessage(&msg, hDlg, 0, 0, PM_REMOVE) )
    {
        TranslateMessage(&msg);
        DispatchMessage(&msg);
    }
    Sleep(250);
    return;
}

```

install.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by install.rc
//
#define IDD_DIALOG1                101
#define IDI_ICON1                  102
#define IDR_TPCCDLL1               103
#define IDD_DIALOG2                104
#define IDI_ICON2                  105
#define IDR_DELIVERY1              109
#define IDD_DIALOG3                110
#define BN_LOG                      1001
#define ED_KEEP                     1002
#define ED_THREADS                  1003
#define ED_THREADS2                 1004
#define ED_MAXWARE                  1006
#define IDC_PATH                    1007
#define IDC_VERSION                 1009
#define IDC_RESULTS                 1010
#define IDC_PROGRESS1              1011
#define IDC_STATUS                  1012
#define IDC_BUTTON1                 1013
#define ED_MAXCONNECTION            1014
#define ED_IIS_MAX_THREAD_POOL_LIMIT 1015
#define ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE 1017
#define ED_IIS_THREAD_TIMEOUT       1018
#define ED_IIS_LISTEN_BACKLOG       1019

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE    111
#define _APS_NEXT_COMMAND_VALUE    40001
#define _APS_NEXT_CONTROL_VALUE    1015
#define _APS_NEXT_SYMED_VALUE      101
#endif
#endif

```

install.rc

```
//Microsoft Developer Studio generated resource script.
//
#include "install.h"

#define APSTUDIO_READONLY_SYMBOLS
////////////////////
//
// Generated from the TEXTINCLUDE 2 resource.
//
#include "afxres.h"

////////////////////
#undef APSTUDIO_READONLY_SYMBOLS

////////////////////
// English (U.S.) resources

#ifdef !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifdef WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // WIN32

////////////////////
//
// Dialog
//

IDD_DIALOG1 DIALOGEX 0, 0, 234, 174
STYLE DS_MODALFRAME | DS_CENTER | WS_MINIMIZEBOX | WS_POPUP | WS_CAPTION |
WS_SYSMENU
CAPTION "TPCC Web Client Installation Utility"
FONT 8, "MS Sans Serif"
BEGIN
    EDITTEXT        ED_MAXWARE,199,37,21,12,ES_NUMBER,WS_EX_RTLREADING
    CONTROL         "",BN_LOG,"Button",BS_AUTOCHECKBOX | BS_LEFTTEXT |
BS_LEFT | BS_VCENTER | WS_TABSTOP,205,51,15,13,
WS_EX_STATICEDGE
    EDITTEXT        ED_THREADS,205,66,15,12,ES_NUMBER,WS_EX_RTLREADING
    EDITTEXT        ED_MAXCONNECTION,186,80,34,12,ES_NUMBER,WS_EX_RTLREADING
    EDITTEXT        ED_IIS_MAX_THREAD_POOL_LIMIT,186,94,34,12,ES_NUMBER,
WS_EX_RTLREADING
    EDITTEXT        ED_WEB_SERVICE_BACKLOG_QUEUE_SIZE,186,108,34,12,
ES_NUMBER,WS_EX_RTLREADING
    EDITTEXT        ED_IIS_THREAD_TIMEOUT,186,122,34,12,ES_NUMBER,
WS_EX_RTLREADING
    EDITTEXT        ED_IIS_LISTEN_BACKLOG,186,136,34,12,ES_NUMBER,
WS_EX_RTLREADING
    DEFPUSHBUTTON   "OK",IDOK,59,153,50,14
    PUSHBUTTON      "Cancel",IDCANCEL,125,153,50,14
    EDITTEXT        IDC_PATH,42,22,178,13,ES_AUTOHSCROLL | ES_READONLY
    LTEXT           "Max Number of Warehouses:",IDC_STATIC,42,37,115,12,
SS_SUNKEN
    LTEXT           "Write HTML To Log file:",IDC_STATIC,42,51,115,12,
SS_SUNKEN
    LTEXT           "Number of Delivery Threads:",IDC_STATIC,42,66,115,12,
SS_SUNKEN
    LTEXT           "Max Number of Connections:",IDC_STATIC,41,80,115,12,
SS_SUNKEN
    CTEXT          "Version 1.00.001",IDC_VERSION,42,6,178,14,SS_SUNKEN |
```

```
WS_BORDER,WS_EX_CLIENTEDGE
    IDI_ICON1,IDC_STATIC,9,6,21,20,0,WS_EX_CLIENTEDGE
    LTEXT           "IIS Max Thread Pool Limit:",IDC_STATIC,41,94,115,12,
SS_SUNKEN
    LTEXT           "Web Service Backlog Queue Size:",IDC_STATIC,41,108,115,
12,SS_SUNKEN
    LTEXT           "IIS Thread Timeout:",IDC_STATIC,41,122,115,12,SS_SUNKEN
    LTEXT           "IIS Listen Backlog:",IDC_STATIC,41,136,115,12,SS_SUNKEN
END

IDD_DIALOG2 DIALOGEX 0, 0, 117, 62
STYLE DS_SETFOREGROUND | DS_3DLOOK | DS_CENTER | WS_POPUP | WS_BORDER
EXSTYLE WS_EX_STATICEDGE
FONT 12, "MS Sans Serif", 0, 0, 0x1
BEGIN
    DEFPUSHBUTTON   "OK",IDOK,33,45,50,9
    CTEXT           "HTML TPCC Installation Successful",IDC_RESULTS,7,22,
102,18,0,WS_EX_CLIENTEDGE
    IDI_ICON2,IDC_STATIC,50,7,18,20,SS_REALSIZEIMAGE,
WS_EX_TRANSPARENT
END

IDD_DIALOG3 DIALOG DISCARDABLE 0, 0, 91, 40
STYLE DS_SYSMODAL | DS_MODALFRAME | DS_3DLOOK | DS_CENTER | WS_CAPTION
CAPTION "Installing TPCC Web Service"
FONT 12, "Arial Black"
BEGIN
    CONTROL         "Progress1",IDC_PROGRESS1,"msctls_progress32",WS_BORDER,
7,20,77,13
    CTEXT           "Static",IDC_STATUS,7,7,77,12,SS_SUNKEN
END

////////////////////
//
// DESIGNINFO
//

#ifdef APSTUDIO_INVOKED
GUIDELINES DESIGNINFO DISCARDABLE
BEGIN
    IDD_DIALOG1, DIALOG
    BEGIN
        LEFTMARGIN, 9
        RIGHTMARGIN, 220
        TOPMARGIN, 6
        BOTTOMMARGIN, 167
    END

    IDD_DIALOG2, DIALOG
    BEGIN
        LEFTMARGIN, 7
        RIGHTMARGIN, 109
        TOPMARGIN, 7
        BOTTOMMARGIN, 54
    END

    IDD_DIALOG3, DIALOG
    BEGIN
        LEFTMARGIN, 7
        RIGHTMARGIN, 84
        TOPMARGIN, 7
        BOTTOMMARGIN, 33
    END
END
#endif
```

```

        END
    END
#endif // APSTUDIO_INVOKED

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// TEXTINCLUDE
//
1 TEXTINCLUDE DISCARDABLE
BEGIN
    "resource.h\0"
END

2 TEXTINCLUDE DISCARDABLE
BEGIN
    "#include "afxres.h"\r\n"
    "\0"
END

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "\r\n"
    "\0"
END

#endif // APSTUDIO_INVOKED

////////////////////////////////////
//
// Icon
//
// Icon with lowest ID value placed first to ensure application icon
// remains consistent on all systems.
IDI_ICON1          ICON    DISCARDABLE    "icon1.ico"
IDI_ICON2          ICON    DISCARDABLE    "icon2.ico"

////////////////////////////////////
//
// TPCCDLL
//
IDR_TPCCDLL1       TPCCDLL DISCARDABLE    "tpcc.dll"

#ifdef _MAC
////////////////////////////////////
//
// Version
//
VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,3,0,2
PRODUCTVERSION 0,3,0,2
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
#endif

```

```

FILEOS 0x40004L
FILETYPE 0x1L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "CompanyName", "Microsoft\0"
            VALUE "FileDescription", "install\0"
            VALUE "FileVersion", "0, 3, 0, 2\0"
            VALUE "InternalName", "install\0"
            VALUE "LegalCopyright", "Copyright © 1996\0"
            VALUE "OriginalFilename", "install.exe\0"
            VALUE "ProductName", "Microsoft install\0"
            VALUE "ProductVersion", "0, 3, 0, 2\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

////////////////////////////////////
//
// DELIVERY
//
IDR_DELIVERY1      DELIVERY DISCARDABLE    "delisrv.exe"
#endif // English (U.S.) resources
////////////////////////////////////

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//
////////////////////////////////////
#endif // not APSTUDIO_INVOKED

```

q.c

```

/*      FILE:          Q.C
*
*      Based on: Microsoft TPC-C Kit Ver. 3.00.000
*
*      Copyright Microsoft, 1996
*      Copyright Performance Tuning Corporation,
1997
*
*      PURPOSE: Tuxedo Server.
*      Original Author:      Philip Durr

```

```

*                               philipdu@Microsoft.com
*
* MODIFIED      Changed for modularity and to allow for the Tuxedo TM
*
* Later Author:      Edward Whalen
*                   Performance Tuning Corporation
*                   ewhalen@perftuning.com
*
* Last Hacker =)
*   Mike Nikolaiev - Changed lots of stuff like pointers to make go fast.
*   Mike Nikolaiev - combined queues in one
*                   Copyright (c) 1998 Compaq Computer Corporation
*/

#include <windows.h>
#include <process.h>
#include <stdio.h>
#include <stdarg.h>
#include <malloc.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <io.h>

#define DBNTWIN32
#include <sqlfront.h>
#include <sqldb.h>

#include "trans.h" //tpckit
transaction header contains definations of structures specific to TPC-C
#include "httpext.h" //ISAPI DLL
information header

#include "tpcc.h" //this dlls
specific structure, value e.t. header.

#include <tmenv.h>
#include <xa.h>
#include <atmi.h>

void NEWORDER();
void PAYMENT();
void ORDERSTATUS();
void STOCKLEVEL();

BOOL      bLog          = FALSE;
BOOL      bFlush       = FALSE;
          //Flush delivery log info when written.
BOOL      verbose      = FALSE;
BOOL      bError       = FALSE;

int       iThreads     = 5;
int       iMaxWareHouses = 500;
int       iDelayMs     = 100;
short    iMaxConnections = (short)1;
short    iDeadlockRetry  = (short)3;

DBPROCESS *pdbproc;

static char          szServer[32];
          //SQL server name

```

```

static char          szDatabase[32];
static char          //tpcc database name          szUser[32];
static char          //user name
static char          //user password          szPassword[32];
int spId;

TERM Term;

static char          szTpccLogPath[256]; //path to html log file if logging turned on
in registry.
static char szErrorLogPath[256]; //path to error log file.

static CRITICAL_SECTION CriticalSection;
static CRITICAL_SECTION ErrorLogCriticalSection;
static EXTENSION_CONTROL_BLOCK *gpECB = NULL;
static int bTpccExit;
          //exit delivery disconnect loop as dll exiting.

extern void TMLog();
extern BOOL SQLInit();
extern void UtilStrCpy();
extern void UtilStrCpy();
extern BOOL SQLOpenConnection();
extern BOOL SQLCloseConnection();
extern BOOL SQLDetectDeadlock();

/* FUNCTION: tpsvrinit ( int argc, char *argv[] )
*
* PURPOSE:      Initialize the Server to Database connection.
*
* RETURNS:      int      0      Success
*               int      -1     Failure
*
* COMMENTS:      None
*/

int tpsvrinit ( int argc, char *argv[] )
{
    if ( GetParameters(argc, argv) )
    {
        PrintParameters();
        return -1;
    }

    if ( verbose )
        TMLog("TPSVRINIT: Q: Server %s, Database %s, User %s, Password
%s, Flush %d.",
            szServer, szDatabase, szUser, szPassword, bFlush);

    if ( ! SQLInit() )
    {
        TMLog( "Q: SQLInit Failed" );
        return -1;
    }
}

```



```

        if ( SQLOpenConnection ( NULL, 0, 0, &pdbproc, szServer, szDatabase,
szUser, szPassword, szDatabase, &spId)
        {
            TMLog ( "Q: SQLOpenConnection Failed" );
            dbexit();
            return -1;
        }
        return 0;
    }

/* FUNCTION: tpsvrdone ( void )
 *
 * PURPOSE:      Initialize the Server to Database connection.
 *
 * RETURNS:      int      0      Success
 *               -1      Failure
 *
 * COMMENTS:     None
 */

void tpsvrdone ( void )
{
    SQLCloseConnection( NULL, pdbproc);
    dbexit();
}

void Q1 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );
            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

void Q2 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );

```

```

            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

void Q3 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );
            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

void Q4 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );
            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

void Q5 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );

```

```

        break;
    case sizeof(ORDER_STATUS_DATA):
        ORDERSTATUS( rqst );
        break;
    case sizeof(STOCK_LEVEL_DATA):
        STOCKLEVEL( rqst );
        break;
    default:
        exit;
    }
}

void Q6 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );
            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

void Q7 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );
            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

void Q8 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );

```

```

        break;
    case sizeof(PAYMENT_DATA):
        PAYMENT( rqst );
        break;
    case sizeof(ORDER_STATUS_DATA):
        ORDERSTATUS( rqst );
        break;
    case sizeof(STOCK_LEVEL_DATA):
        STOCKLEVEL( rqst );
        break;
    default:
        exit;
    }
}

void Q9 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );
            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

void Q10 ( TPSVCINFO *rqst )
{
    int size = rqst->len;
    switch(size)
    {
        case sizeof(NEW_ORDER_DATA):
            NEWORDER( rqst );
            break;
        case sizeof(PAYMENT_DATA):
            PAYMENT( rqst );
            break;
        case sizeof(ORDER_STATUS_DATA):
            ORDERSTATUS( rqst );
            break;
        case sizeof(STOCK_LEVEL_DATA):
            STOCKLEVEL( rqst );
            break;
        default:
            exit;
    }
}

/* FUNCTION: NEWORDER ( TPSVCINFO *rqst )
 *
 * PURPOSE:      Process a New Order request.
 *

```

```

* RETURNS:          int      0          Success
*                  *         -1         Failure
*
*
* COMMENTS:        None
*
*/

void NEWORDER ( TPSVCINFO *rqst )
{
    PECBINFO pECBInfo = dbgetuserdata(pdbproc);
    NEW_ORDER_DATA *np;
    int size = rqst->len;

    np = (NEW_ORDER_DATA * ) rqst->data;

    if (verbose )
    {
        TMLog(" NEWORDER: d_id %d ", np->d_id);
        TMLog(" NEWORDER: c_id %d ", np->c_id);
        TMLog(" NEWORDER: o_ol_cnt %d ", np->o_ol_cnt);
        TMLog(" NEWORDER: o_all_local %d ", np->o_all_local);
    }

    bError = FALSE;

    np->retval = SQLNewOrder( NULL, 0, 0, pdbproc, np, iDeadlockRetry);

    if (bError == TRUE)
        np->retval = -1;

    if (verbose )
        TMLog(" NEWORDER: Return Value %d", np->retval);

    tpreturn( TPSUCCESS, 0, rqst->data, size, 0);
}

/* FUNCTION: int SQLNewOrder(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId, int iTermId, int iSyncId, DBPROCESS *dbproc, NEW_ORDER_DATA *pNewOrder,
short deadlock_retry)
*
* PURPOSE:        This function handles the new order transaction.
*
* ARGUMENTS:     EXTENSION_CONTROL_BLOCK *pECB
    passed in structure pointer from inetsrv.
*
*                int
*                iTermId          terminal id of browser
*
*                int
*                iSyncId         sync id of browser
*
*                DBPROCESS
*                *dbproc         connection db process id
*
*                NEW_ORDER_DATA
*                *pNewOrder      pointer to new order structure for
input/output data
*
*                short
*                deadlock_retry  retry count if deadlocked
*
* RETURNS:       int      TRUE      transaction committed
*                *         FALSE     item number not valid
*                *         -1       deadlock max
*
* retry reached
*

```

```

*
* COMMENTS:      None
*
*/

static int SQLNewOrder(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId,
DBPROCESS *dbproc, NEW_ORDER_DATA *pNewOrder, short deadlock_retry)
{
    RETCODE          rc;
    int              i;
    DBINT            commit_flag;
    int              tryit;
    char             printbuf[25];
    char             tmpbuf[30];
    DBDATETIME       datetime;
    BYTE             *pData;
    PECBINFO         pEcbInfo;

    if ( (pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
    {
        pEcbInfo->pECB = pECB;
        pEcbInfo->bFailed = FALSE;
        pEcbInfo->iTermId = iTermId;
        pEcbInfo->iSyncId = iSyncId;
    }

    pNewOrder->num_deadlocks = 0;

    strcpy(tmpbuf, "tpcc_neworder");

    for (tryit=0; tryit < deadlock_retry; tryit++)
    {
        if (dbrpcinit(dbproc, tmpbuf, 0) == SUCCEED)
        {
            dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1, (BYTE *)
&pNewOrder->w_id);
            dbrpcparam(dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&pNewOrder->d_id);
            dbrpcparam(dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&pNewOrder->c_id);
            dbrpcparam(dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&pNewOrder->o_ol_cnt);
            dbrpcparam(dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&pNewOrder->o_all_local);

            for (i = 0; i < pNewOrder->o_ol_cnt; i++)
            {
                dbrpcparam(dbproc, NULL, 0, SQLINT4, -1, -1,
(BYTE *) &pNewOrder->ol[i].ol_i_id);
                dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1,
(BYTE *) &pNewOrder->ol[i].ol_supply_w_id);
                dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1,
(BYTE *) &pNewOrder->ol[i].ol_quantity);
            }

            if (dbrpcexec(dbproc) == SUCCEED)
            {
                pNewOrder->total_amount=0;

                // Get results from order line
                for (i = 0; i<pNewOrder->o_ol_cnt; i++)
                {

```

```

        if ((rc = dbresults(dbproc)) !=
NO_MORE_RESULTS) && (rc != FAIL))
    {
        if (DBROWS(dbproc) &&
(dbnumcols(dbproc) == 5))
            while
            {
                if (pData=dbdata(dbproc, 1))
                UtilStrCpy(pNewOrder->ol[i].ol_i_name, pData, dbdatlen(dbproc, 1));
                if (pData=dbdata(dbproc, 2))
                pNewOrder->ol[i].ol_stock = *(DBSMALLINT *) pData);
                if (pData=dbdata(dbproc, 3))
                UtilStrCpy(pNewOrder->ol[i].ol_brand_generic, pData, dbdatlen(dbproc, 3));
                if (pData=dbdata(dbproc, 4))
                pNewOrder->ol[i].ol_i_price = *(DBFLT8 *) pData);

                if (pData=dbdata(dbproc, 5))

                pNewOrder->ol[i].ol_amount = *(DBFLT8 *) pData);

                pNewOrder->total_amount = pNewOrder->total_amount + pNewOrder-
>ol[i].ol_amount;
            }
        }
    }
    while ((rc = dbresults(dbproc)) !=
NO_MORE_RESULTS) && (rc != FAIL))
    {
        if (DBROWS(dbproc) &&
(dbnumcols(dbproc) == 8))
            while ((rc =
dbnextrow(dbproc)) != NO_MORE_ROWS) && (rc != FAIL))
            {
                if (pData=dbdata(dbproc, 1))

                pNewOrder->w_tax = *(DBFLT8 *) pData);
            }
    }

```

```

        if (pData=dbdata(dbproc, 2))

        pNewOrder->d_tax = *(DBFLT8 *) pData);

        if (pData=dbdata(dbproc, 3))
        pNewOrder->o_id = *(DBINT *) pData);
        if (pData=dbdata(dbproc, 4))
        UtilStrCpy(pNewOrder->c_last, pData, dbdatlen(dbproc, 4));
        if (pData=dbdata(dbproc, 5))
        pNewOrder->c_discount = *(DBFLT8 *) pData);

        if (pData=dbdata(dbproc, 6))
        UtilStrCpy(pNewOrder->c_credit, pData, dbdatlen(dbproc, 6));
        if (pData=dbdata(dbproc, 7))
        {
            datetime = *((DBDATETIME *) pData);
            dbdatecrack(dbproc, &pNewOrder->o_entry_d, &datetime);
        }
        if (pData=dbdata(dbproc, 8)) commit_flag = *(DBTINYINT *) pData);
    }
}
}
}
if (SQLDetectDeadlock(dbproc))
{
    pNewOrder->num_deadlocks++;
    sprintf(printbuf, "deadlock: retry: %d", pNewOrder-
>num_deadlocks);
    Sleep(DEADLOCKWAIT*tryit);
}
else
{
    if (commit_flag == 1)
    {
        pNewOrder->total_amount = pNewOrder-
>total_amount * ((1 + pNewOrder->w_tax + pNewOrder->d_tax) * (1 - pNewOrder-
>c_discount));
        strcpy(pNewOrder-
>execution_status, "Transaction committed.");
        return TRUE;
    }
}

```

```

        }
        else
        {
            strcpy(pNewOrder->execution_status,"Item
number is not valid.");
            return FALSE;
        }
    }
}

// If we reached here, it means we quit after MAX_RETRY deadlocks
strcpy(pNewOrder->execution_status,"Hit deadlock max. ");
if ( verbose )
    TMLog(" NEWORDER: SQLNewOrder Max Deadlocks %d", tryit);

return -1; // "deadlock max retry reached!"
}

/*****
*****/

/* FUNCTION: PAYMENT ( TPSVCINFO *rqst )
*
* PURPOSE:      Process a Payment request.
*
* RETURNS:      int      0      Success
*               -1      Failure
*
* COMMENTS:     None
*
*/

void PAYMENT ( TPSVCINFO *rqst )
{
    PECBINFO pECBInfo = dbgetuserdata(pdbproc);
    PAYMENT_DATA *pp;
    int size = rqst->len;

    pp = (PAYMENT_DATA *) rqst->data ;

    if (verbose )
    {
        TMLog(" PAYMENT: w_id %d ", pp->w_id);
        TMLog(" PAYMENT: d_id %d ", pp->d_id);
        TMLog(" PAYMENT: c_id %d ", pp->c_id);
    }

    bError = FALSE;

    pp->retval = SQLPayment( NULL, 0, 0, pdbproc, pp, iDeadlockRetry);

    if (bError == TRUE)
        pp->retval = -1;

    if (verbose )
        TMLog(" PAYMENT: Return Value %d", pp->retval);

    tpreturn( TPSUCCESS, 0, rqst->data, size, 0);
}

```

```

/* FUNCTION: int SQLPayment(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId,
DBPROCESS *dbproc, PAYMENT_DATA *pPayment, short deadlock_retry)
*
* PURPOSE:      This function handles the payment transaction.
*
* ARGUMENTS:    EXTENSION_CONTROL_BLOCK *pECB
                passed in structure pointer from inetsrv.
*               int
                iTermId      terminal id of browser
*               int
                iSyncId     sync id of browser
*               DBPROCESS
                *dbproc      connection db process id
*               PAYMENT_DATA
                *pPayment    pointer to payment input/output data structure
                short
                deadlock_retry deadlock retry count
*
* RETURNS:      int      TRUE      success
*               -1      success
                max
                deadlocked reached
*
* COMMENTS:     None
*
*/

static int SQLPayment(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId,
DBPROCESS *dbproc, PAYMENT_DATA *pPayment, short deadlock_retry)
{
    RETCODE      rc;
    int          tryit;
    char         printbuf[26];
    BOOL         by_name;
    DBDATETIME  datetime;
    BYTE         *pData;
    PECBINFO     pEcbInfo;

    if ( (pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
    {
        pEcbInfo->pECB = pECB;
        pEcbInfo->bFailed = FALSE;
        pEcbInfo->iTermId = iTermId;
        pEcbInfo->iSyncId = iSyncId;
    }

    pPayment->num_deadlocks = 0;

    if (pPayment->c_id == 0)
        by_name = TRUE;
    else
        by_name = FALSE;

    for (tryit=0; tryit < deadlock_retry; tryit++)
    {
        if (dbrpcinit(dbproc, "tpcc_payment", 0) == SUCCEEDED)
        {
            dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1, (BYTE *)
&pPayment->w_id);
            dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1, (BYTE *)
&pPayment->c_w_id);
            dbrpcparam(dbproc, NULL, 0, SQLFLT8, -1, -1, (BYTE *)
&pPayment->h_amount);

```

```

        dbrpcparam(dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&pPayment->d_id);
        dbrpcparam(dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&pPayment->c_d_id);
        dbrpcparam(dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&pPayment->c_id);
        if (pPayment->c_id == 0)
        {
            dbrpcparam(dbproc, NULL, 0, SQLCHAR, -1,
strlen(pPayment->c_last), pPayment->c_last);
        }
        if (dbrpcexec(dbproc) == SUCCEED)
        {
            while (((rc = dbresults(dbproc)) != NO_MORE_RESULTS)
&& (rc != FAIL))
            {
                if (DBROWS(dbproc) && (dbnumcols(dbproc) ==
27))
                {
                    while (((rc = dbnextrow(dbproc))
!= NO_MORE_ROWS) && (rc != FAIL))
                    {
                        if (pData=dbdata(dbproc,
1))
                            pPayment->c_id
= *((DBINT *) pData);
                        if (pData=dbdata(dbproc,
2))
                            UtilStrCpy(pPayment->c_last, pData, dbdatlen(dbproc, 2));
                        if (pData=dbdata(dbproc,
3))
                            {
                                datetime =
*(DBDATETIME *) pData);
                                dbdatecrack(dbproc, &pPayment->h_date, &datetime);
                            }
                        if (pData=dbdata(dbproc,
4))
                            UtilStrCpy(pPayment->w_street_1, pData, dbdatlen(dbproc, 4));
                        if (pData=dbdata(dbproc,
5))
                            UtilStrCpy(pPayment->w_street_2, pData, dbdatlen(dbproc, 5));
                        if (pData=dbdata(dbproc,
6))
                            UtilStrCpy(pPayment->w_city, pData, dbdatlen(dbproc, 6));
                        if (pData=dbdata(dbproc,
7))
                            UtilStrCpy(pPayment->w_state, pData, dbdatlen(dbproc, 7));
                        if (pData=dbdata(dbproc,
8))
                            UtilStrCpy(pPayment->w_zip, pData, dbdatlen(dbproc, 8));
                        if (pData=dbdata(dbproc,
9))
                            UtilStrCpy(pPayment->d_street_1, pData, dbdatlen(dbproc, 9));

```

```

10)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->d_street_2, pData, dbdatlen(dbproc, 10));
11)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->d_city, pData, dbdatlen(dbproc, 11));
12)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->d_state, pData, dbdatlen(dbproc, 12));
13)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->d_zip, pData, dbdatlen(dbproc, 13));
14)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_first, pData, dbdatlen(dbproc, 14));
15)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_middle, pData, dbdatlen(dbproc, 15));
16)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_street_1, pData, dbdatlen(dbproc, 16));
17)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_street_2, pData, dbdatlen(dbproc, 17));
18)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_city, pData, dbdatlen(dbproc, 18));
19)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_state, pData, dbdatlen(dbproc, 19));
20)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_zip, pData, dbdatlen(dbproc, 20));
21)
        if (pData=dbdata(dbproc,
UtilStrCpy(pPayment->c_phone, pData, dbdatlen(dbproc, 21));
22)
        if (pData=dbdata(dbproc,
        {
            datetime =
*(DBDATETIME *) pData);
            dbdatecrack(dbproc, &pPayment->c_since, &datetime);
        }
        if (pData=dbdata(dbproc,
23)
        UtilStrCpy(pPayment->c_credit, pData, dbdatlen(dbproc, 23));
        if (pData=dbdata(dbproc,
24)

```

```

                pPayment-
>c_credit_lim = (*(DBFLT8 *) pData);
25)                if (pData=dbdata(dbproc,
                pPayment-
>c_discount = (*(DBFLT8 *) pData);
26)                if (pData=dbdata(dbproc,
                pPayment-
>c_balance = (*(DBFLT8 *) pData);
27)                if (pData=dbdata(dbproc,

                UtilStrCpy(pPayment->c_data, pData, dbdatlen(dbproc, 27));
                }
            }
        }
        if (SQLDetectDeadlock(dbproc))
        {
            pPayment->num_deadlocks++;
            sprintf(printbuf, "deadlock: retry: %d", pPayment-
>num_deadlocks);
            Sleep(DEADLOCKWAIT*tryit);
        }
        else
        {
            if ( pPayment->c_id == 0 )
            {
                strcpy(pPayment->execution_status, "Invalid
Customer id,name.");
                return 0;
            }
            else
                strcpy(pPayment-
>execution_status, "Transaction committed.");
            return TRUE;
        }
    }
    // If we reached here, it means we quit after MAX_RETRY deadlocks
    strcpy(pPayment->execution_status, "Hit deadlock max. ");
    return -1; /*"deadlock max retry reached!"
}

/*****
*****/

void STOCKLEVEL ( TPSVCINFO *rqst )
{
    PECBINFO pECBInfo = dbgetuserdata(pdbproc);
    STOCK_LEVEL_DATA *sp;
    int size = rqst->len;

    sp = (STOCK_LEVEL_DATA *) rqst->data;

    if (verbose )
    {
        TMLog(" STOCKLEVEL: w_id %d ", sp->w_id);
        TMLog(" STOCKLEVEL: d_id %d ", sp->d_id);
        TMLog(" STOCKLEVEL: c_id %d ", sp->thresh_hold);
    }

    bError = FALSE;

```

```

    sp->retval = SQLStockLevel( NULL, 0, 0, pdbproc, sp, iDeadlockRetry);

    if (bError == TRUE)
        sp->retval = -1;

    if ( verbose )
        TMLog(" STOCKLEVEL: Return Value %d", sp->retval);

    tpreturn( TPSUCCESS, 0, rqst->data, size, 0);
}

/* FUNCTION: SQLStockLevel(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId, DBPROCESS *dbproc, STOCK_LEVEL_DATA *pStockLevel, short deadlock_retry)
*
* PURPOSE:      This function handles the stock level transaction.
*
* ARGUMENTS:    EXTENSION_CONTROL_BLOCK *pECB
                passed in structure pointer from inetsrv.
*
                iTermId          int          terminal id of browser
*
                iSyncId          int          sync id of browser
*
                *dbproc          DBPROCESS    connection db process id
                STOCK_LEVEL_DATA *pStockLevel
                stock level input / output data structure
*
                deadlock_retry  short        retry count if deadlocked
*
* RETURNS:      BOOL          FALSE          if successfull
                TRUE          if
deadlocked
*
* COMMENTS:     None
*/

static BOOL SQLStockLevel(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId,
DBPROCESS *dbproc, STOCK_LEVEL_DATA *pStockLevel, short deadlock_retry)
{
    int          tryit;
    RETCODE      rc;
    char         printbuf[25];
    BYTE         *pData;
    PECBINFO     pEcbInfo;

    //update pECB and bFailed flag
    if ( (pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
    {
        pEcbInfo->pECB = pECB;
        pEcbInfo->bFailed = FALSE;
        pEcbInfo->iTermId = iTermId;
        pEcbInfo->iSyncId = iSyncId;
    }

    pStockLevel->num_deadlocks = 0;

    for (tryit=0; tryit < deadlock_retry; tryit++)
    {
        if (dbrpcinit(dbproc, "tpcc_stocklevel", 0) == SUCCEED)
        {

```

```

        dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1, (BYTE *)
&pStockLevel->w_id);
        dbrpcparam(dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&pStockLevel->d_id);
        dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1, (BYTE *)
&pStockLevel->thresh_hold);

        if (dbrpcexec(dbproc) == SUCCEED)
        {
            while ((rc = dbresults(dbproc)) !=
NO_MORE_RESULTS) && (rc != FAIL))
            {
                if (DBROWS(dbproc))
                {
                    while ((rc =
dbnextrow(dbproc)) != NO_MORE_ROWS) && (rc != FAIL))
                    {
                        if (pData=dbdata(dbproc, 1))
                        pStockLevel->low_stock = *((long *) pData);
                    }
                }
            }
            if (SQLDetectDeadlock(dbproc))
            {
                pStockLevel->num_deadlocks++;
                sprintf(printbuf,"deadlock: retry: %d",pStockLevel-
>num_deadlocks);
                Sleep(10 * tryit);
            }
            else
            {
                strcpy(pStockLevel->execution_status, "Transaction
committed.");
                return TRUE;
            }
        }

        // If we reached here, it means we quit after MAX_RETRY deadlocks
        strcpy(pStockLevel->execution_status, "Hit deadlock max. ");
        return -1;
    }

/*****
*****/

/* FUNCTION: ORDERSTATUS ( TPSVCINFO *rqst )
*
* PURPOSE:      Process an Order Status request.
*
* RETURNS:      int      0      Success
*               int      -1     Failure
*
* COMMENTS:     None
*
*/

```

```

void ORDERSTATUS ( TPSVCINFO *rqst )
{
    PECBINFO pECBInfo = dbgetuserdata(pdbproc);
    ORDER_STATUS_DATA *op ;
    int size = rqst->len;

    op = (ORDER_STATUS_DATA *) rqst->data;

    if (verbose )
    {
        TMLog(" ORDERSTATUS: w_id %d ", op->w_id);
        TMLog(" ORDERSTATUS: d_id %d ", op->d_id);
        TMLog(" ORDERSTATUS: c_id %d ", op->c_id);
    }

    bError = FALSE;

    op->retval = SQLOrderStatus( NULL, 0, 0, pdbproc, op, iDeadlockRetry);

    if (bError == TRUE)
        op->retval = -1;

    if ( verbose )
        TMLog(" ORDERSTATUS: Return Value %d", op->retval);

    if (verbose )
    {
        TMLog(" ORDERSTATUS: w_id %d ", op->w_id);
        TMLog(" ORDERSTATUS: d_id %d ", op->d_id);
        TMLog(" ORDERSTATUS: c_id %d ", op->c_id);
        TMLog(" ORDERSTATUS: c_balance %f ", op->c_balance);
        TMLog(" ORDERSTATUS: o_id %d ", op->o_id);
        TMLog(" ORDERSTATUS: o_carrier_id %d ", op->o_carrier_id);
        TMLog(" ORDERSTATUS: execution_status %s ", op-
>execution_status);
    }

    tpreturn( TFSUCCESS, 0, rqst->data, size, 0);
}

/* FUNCTION: int SQLOrderStatus(EXTENSION CONTROL_BLOCK *pECB, int iTermId, int
iSyncId, DBPROCESS *dbproc, ORDER_STATUS_DATA *pOrderStatus, short deadlock_retry)
*
* PURPOSE:      This function processes the Order Status transaction.
*
* ARGUMENTS:    EXTENSION_CONTROL_BLOCK      *pECB
                passed in structure pointer from inetrv.
*
                iTermId                      int          terminal id of browser
*
                iSyncId                      int          sync id of browser
*
                *dbproc                      DBPROCESS    connection db process id
                ORDER_STATUS_DATA           *pOrderStatus
                pointer to Order Status data input/output structure
*
                deadlock_retry               short
                deadlock_retry count
*
* RETURNS:      int      -1      max deadlock reached
*               int      0       No orders
                found for customer
*
                int      1       Transaction
                successfull

```



```

*
* COMMENTS:      None
*
*/

static int SQLOrderStatus(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId,
DBPROCESS *dbproc, ORDER_STATUS_DATA *pOrderStatus, short deadlock_retry)
{
    RETCODE          rc;
    int               tryit;
    int               i;
    char              printbuf[25];
    BOOL              by_name;
    DBDATETIME        datetime;
    BYTE              *pData;
    PECBINFO          pEcbInfo;

    if ( (pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
    {
        pEcbInfo->pECB = pECB;
        pEcbInfo->bFailed = FALSE;
        pEcbInfo->iTermId = iTermId;
        pEcbInfo->iSyncId = iSyncId;
    }

    pOrderStatus->num_deadlocks = 0;
    if (pOrderStatus->c_id == 0)
        by_name = TRUE;
    else
        by_name = FALSE;

    for (tryit=0; tryit < deadlock_retry; tryit++)
    {
        if (dbrpcinit(dbproc, "tpcc_orderstatus", 0) == SUCCEED)
        {
            dbrpcparam(dbproc, NULL, 0, SQLINT2, -1, -1, (BYTE *)
&pOrderStatus->w_id);
            dbrpcparam(dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&pOrderStatus->d_id);
            dbrpcparam(dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&pOrderStatus->c_id);
            if (pOrderStatus->c_id == 0)
            {
                dbrpcparam(dbproc, NULL, 0, SQLCHAR, -1,
strlen(pOrderStatus->c_last), pOrderStatus->c_last);
            }
            if (dbrpcexec(dbproc) == SUCCEED)
            {
                while (((rc = dbresults(dbproc)) != NO_MORE_RESULTS)
&& (rc != FAIL))
                {
                    if (DBROWS(dbproc) && (dbnumcols(dbproc) ==
5))
                    {
                        i=0;
                        while (((rc = dbnextrow(dbproc))
!= NO_MORE_ROWS) && (rc != FAIL))
                        {
                            if (pData=dbdata(dbproc,
pOrderStatus-
>O1OrderStatusData[i].ol_supply_w_id = (*(DBSMALLINT *) pData);

```

```

                if (pData=dbdata(dbproc,
pOrderStatus-
>O1OrderStatusData[i].ol_i_id = (*(DBINT *) pData);
                if (pData=dbdata(dbproc,
pOrderStatus-
>O1OrderStatusData[i].ol_quantity = (*(DBSMALLINT *) pData);
                if (pData=dbdata(dbproc,
pOrderStatus-
>O1OrderStatusData[i].ol_amount = (*(DBFLT8 *) pData);
                if (pData=dbdata(dbproc,
{
                    datetime =
*(DBDATETIME *) pData);
                    dbdatecrack(dbproc, &pOrderStatus->O1OrderStatusData[i].ol_delivery_d,
&datetime);
                }
                i++;
            }
            pOrderStatus->o_ol_cnt = i;
        }
        else if (DBROWS(dbproc) &&
(dbnumcols(dbproc) == 8))
        {
            while (((rc = dbnextrow(dbproc))
!= NO_MORE_ROWS) && (rc != FAIL))
            {
                if (pData=dbdata(dbproc,
pOrderStatus-
>c_id = (*(DBINT *) pData);
                if (pData=dbdata(dbproc,
UtilStrCpy(pOrderStatus->c_last, pData, dbdatlen(dbproc,2));
                if (pData=dbdata(dbproc,
UtilStrCpy(pOrderStatus->c_first, pData, dbdatlen(dbproc,3));
                if (pData=dbdata(dbproc,
UtilStrCpy(pOrderStatus->c_middle, pData, dbdatlen(dbproc,4));
                if (pData=dbdata(dbproc,
{
                    datetime =
*(DBDATETIME *) pData);
                    dbdatecrack(dbproc, &pOrderStatus->o_entry_d, &datetime);
                }
                if (pData=dbdata(dbproc,
pOrderStatus-
>o_carrier_id = (*(DBSMALLINT *) pData);
                if (pData=dbdata(dbproc,
pOrderStatus-
>c_balance = (*(DBFLT8 *) pData);

```

```

                                if (pData=dbdata(dbproc,
8))
                                pOrderStatus-
>o_id = (*(DBINT *) pData);
                                }
                                if (i==0)
customer"
                                return 0; //"No orders found for
                                }
                                if (SQLDetectDeadlock(dbproc))
                                {
                                pOrderStatus->num_deadlocks++;
                                sprintf(printbuf,"deadlock: retry: %d",pOrderStatus-
>num_deadlocks);
                                Sleep(DEADLOCKWAIT*tryit);
                                }
                                else
                                {
                                if (pOrderStatus->c_id == 0 && pOrderStatus->c_last[0]
== 0)
                                strcpy(pOrderStatus-
>execution_status,"Invalid Customer id,name.");
                                else
                                strcpy(pOrderStatus-
>execution_status,"Transaction committed.");
                                return 1;
                                }
                                }
                                // If we reached here, it means we quit after MAX_RETRY deadlocks
                                strcpy(pOrderStatus->execution_status,"Hit deadlock max. ");
                                return -1; //"deadlock max retry reached!"
                                }

/*****
*****/

/*
 * Common Code for all Servers
 */

/* FUNCTION: BOOL SQLInit()
 * PURPOSE: This function initializes SQL Server for later use.
 * RETURNS: BOOL FALSE if successfull
 *          TRUE if an error
occurs and connection cannot be established.
 * COMMENTS: None
 */
BOOL SQLInit ()
{
    dbinit();

    if ( dbgetmaxprocs() < iMaxConnections )
    {

```

```

                                if ( dbsetmaxprocs(iMaxConnections) == FAIL )
                                {
                                //set for fail error message when HttpExtensionProc()
is called because
                                //at this point we don't have a pECB so no way to show
error message.
                                iMaxConnections = -1;
                                }
                                }

                                // install error and message handlers
                                dbmsghandle((DBMSGHANDLE_PROC)msg_handler);
                                dberrhandle((DBERRHANDLE_PROC)err_handler);

                                return TRUE;
                                }

/* FUNCTION: BOOL SQLOpenConnection(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId, DBPROCESS **dbproc, char *server, char *database, char *user, char
*password, char *app, int *spid, long *pack_size)
 *
 * PURPOSE: This function opens the sql connection for use.
 * ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
 *
 *          int
 *          iTermId terminal id of browser
 *          int
 *          iSyncId sync id of browser
 *          DBPROCESS **dbproc pointer to
returned DBPROCESS
 *          char
 *          *server
 *          char
 *          *database SQL
 *          char
 *          *user
 *          char
 *          *password user
 *          char
 *          *app
 *          int
 *          *spid pointer to returned spid
 *          long
 *          *pack_size
 *          pointer to returned default pack size
 * RETURNS: BOOL FALSE if successfull
 *          TRUE if an error
occurs
 * COMMENTS: None
 */

#ifdef USE_ODBC
static BOOL SQLOpenConnection(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
int iSyncId, DBPROCESS **dbproc, char *server, char *database, char *user, char
*password, char *app, int *spid, long *pack_size)
{
    RETCODE rc;

```

```

char    buffer[30];

*dbproc = (DBPROCESS *)malloc(sizeof(DBPROCESS));
if ( !*dbproc )
    return TRUE;

//set pECB data into dbproc
(*dbproc)->bDeadlock = FALSE;
(*dbproc)->bFailed = FALSE;
(*dbproc)->pECB = pECB;
(*dbproc)->iTermId = iTermId;
(*dbproc)->iSyncId = iSyncId;

if ( SQLAllocConnect(henv, &(*dbproc)->hdbc) == SQL_ERROR )
    return TRUE;

if ( SQLSetConnectOption((*dbproc)->hdbc, SQL_PACKET_SIZE,
pack_size) == SQL_ERROR )
    return TRUE;

rc = SQLConnect((*dbproc)->hdbc, server, SQL_NTS, user, SQL_NTS,
password, SQL_NTS);
if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
    return TRUE;
rc = SQLAllocStmnt((*dbproc)->hdbc, &(*dbproc)->hstmt);
if (rc == SQL_ERROR)
    return TRUE;

sprintf(buffer,"use %s", Client->database);

rc = SQLExecDirect((*dbproc)->hstmt, buffer, SQL_NTS);
if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
    return TRUE;

SQLFreeStmnt((*dbproc)->hstmt, SQL_CLOSE);
sprintf(buffer,"set nocount on");
rc = SQLExecDirect((*dbproc)->hstmt, buffer, SQL_NTS);
if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
    return TRUE;
SQLFreeStmnt((*dbproc)->hstmt, SQL_CLOSE);

sprintf(buffer,"select @@spid");

rc = SQLExecDirect((*dbproc)->hstmt, buffer, SQL_NTS);
if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
    return TRUE;

if ( SQLBindCol((*dbproc)->hstmt, 1, SQL_C_SSHORT, &(*dbproc)-
>spid, 0, NULL) == SQL_ERROR )
    return TRUE;

if ( SQLFetch((*dbproc)->hstmt) == SQL_ERROR )
    return TRUE;

SQLFreeStmnt((*dbproc)->hstmt, SQL_CLOSE);

return FALSE;
}

#else

```

```

static BOOL SQLOpenConnection(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
int iSyncId, DBPROCESS **dbproc, char *server, char *database, char *user, char
*password, char *app, int *spid)
{
    LOGINREC *login;
    PECBINFO pEcbInfo;

    //set local msg proc for login record
    //attach pECB record

    //this is necessary as dblib provides no way to pass user data
    in a login structure. So until
    //there is an allocated dbproc we need to use a static which
    means that the login attempt must
    //be serialized.

    gpECB = pECB;

    login = dblogin();
    if ( !*user )
        DBSETLUSER(login, "sa");
    else
        DBSETLUSER(login, user);

    DBSETLPWD(login, password);
    DBSETLHOST(login, app);

    DBSETLPACKET(login, (unsigned short)DEFCLPACKSIZE);

    if ((*dbproc = dbopen(login, server)) == NULL)
        return TRUE;

    //set pECB data into dbproc
    pEcbInfo = (PECBINFO)malloc(sizeof(ECBINFO));
    pEcbInfo->bDeadlock = FALSE;
    pEcbInfo->pECB = pECB;
    pEcbInfo->iTermId = iTermId;
    pEcbInfo->iSyncId = iSyncId;
    dbsetuserdata(*dbproc, pEcbInfo);

    // Use the the right database
    dbuse(*dbproc, database);

    dbcmd(*dbproc, "select @@spid");

    dbsqlxec(*dbproc);
    while (dbresults(*dbproc) != NO_MORE_RESULTS)
    {
        dbbind(*dbproc, 1, SMALLBIND, (DBINT) 0, (BYTE *)
spid);

        while (dbnextrow(*dbproc) != NO_MORE_ROWS)
            ;
    }
    dbcmd(*dbproc, "set nocount on");

    dbsqlxec(*dbproc);
    while (dbresults(*dbproc) != NO_MORE_RESULTS)
    {
        while (dbnextrow(*dbproc) != NO_MORE_ROWS)
            ;
    }

    //rollback transaction on abort

```

```

        dbcmd(*dbproc, "set XACT_ABORT ON");

        dbsqlxec(*dbproc);
        while (dbresults(*dbproc) != NO_MORE_RESULTS)
        {
                while (dbnextrow(*dbproc) != NO_MORE_ROWS)
                ;
        }

        return FALSE;
}

#endif

/* FUNCTION: BOOL SQLCloseConnection(EXTENSION_CONTROL_BLOCK *pECB, DBPROCESS
*dbproc)
*
* PURPOSE:      This function closes the sql connection.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*
*               DBPROCESS *dbproc pointer to
DBPROCESS
*
* RETURNS:     BOOL FALSE if successfull
*               TRUE if an error
occurs
*
* COMMENTS:    None
*
*/

#ifdef USE_ODBC
static BOOL SQLCloseConnection(EXTENSION_CONTROL_BLOCK *pECB, DBPROCESS
*dbproc)
{
        if ( dbproc )
        {
                SQLFreeStmt(dbproc->hstmt, SQL_DROP);
                SQLDisconnect(dbproc->hdbc);
                SQLFreeConnect(dbproc->hdbc);
                free(dbproc);
                dbproc = NULL;
        }
        return FALSE;
}

#else
static BOOL SQLCloseConnection(EXTENSION_CONTROL_BLOCK *pECB, DBPROCESS
*dbproc)
{
        if (dbclose(dbproc) == FAIL)
                return TRUE;
        return FALSE;
}

#endif

/* FUNCTION: BOOL SQLDetectDeadlock(DBPROCESS *dbproc)
*
* PURPOSE:      This function checks to see if a sql server deadlock condition
exists.
*
* ARGUMENTS:   DBPROCESS *dbproc
connection db process id to check

```

```

*
* RETURNS:     BOOL FALSE no deadlock detected
*               TRUE
deadlock condition exists
*
* COMMENTS:    None
*
*/

BOOL SQLDetectDeadlock(DBPROCESS *dbproc)
{
        PECBINFO pEcbInfo;

        if ( (pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
        {
                if ( pEcbInfo->bDeadlock )
                {
                        pEcbInfo->bDeadlock = FALSE;
                        return TRUE;
                }
        }
        return FALSE;
}

// Lifted from HP FDR since they did such a nice job
void TMLog( char *format, ... )
{
        va_list args;
        char buf[4096];
        int len;
        va_start( args, format );
        _strtime( buf );
        strcat( buf, " ");
        len = strlen( buf );
        (void)_vsprintf( buf+ len, sizeof( buf ) - len - 1, format, args);
        buf[sizeof( buf ) - 1] = '\0';
        va_end( args );
        userlog( buf );
}

/* FUNCTION: void UtilStrCpy(char *pDest, char *pSrc, int n)
*
* PURPOSE:      This function copies n characters from string pSrc to pDst and
places a
*
*               null character at the end of the destination string.
*
* ARGUMENTS:   char *pDest destination string
pointer
*
*               char *pSrc
source string pointer
*
*               int n
number of characters to copy
*
* RETURNS:     None
*
* COMMENTS:    Unlike strncpy this function ensures that the result string is
always null terminated.
*
*/

static void UtilStrCpy(char *pDest, char *pSrc, int n)
{
        strncpy(pDest, pSrc, n);
}

```

```

    pDest[n] = '\0';
        return;
    }

/* FUNCTION: int err_handler(DBPROCESS *dbproc, int severity, int dberr, int oserr,
char *dberrstr, char *oserrstr)
*
* PURPOSE:      This function handles DB-Library errors
*
* ARGUMENTS:   DBPROCESS      *dbproc          DBPROCESS id
pointer
*              severity       severity of error
*              dberr          int              error id
*              oserr          int              operating system specific error code
*              printable error description of dberr char *dberrstr
*              printable error description of oserr char *oserrstr
*
* RETURNS:     int              INT_CONTINUE
*              continue if error is SQLETIME else INT_CANCEL action
*
* COMMENTS:    None
*/

```

```

int err_handler(DBPROCESS *dbproc, int severity, int dberr, int oserr, char
*dberrstr, char *oserrstr)
{
    PECBINFO
    EXTENSION_CONTROL_BLOCK *pECB;    pEcbInfo;
    FILE *fp;
    SYSTEMTIME systemTime;
    char szTmp[256];
    int iTermId;
    int iSyncId;

    pEcbInfo = NULL;

    if ((dbproc == NULL) || (DBDEAD(dbproc)))
    {
        TMLog("DBPROC is invalid");
        return INT_CANCEL;
    }

    if (!(pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)))
    {
        pECB = gpECB;
        iTermId = 0;
        iSyncId = 0;
    }
    else
    {
        pECB = pEcbInfo->pECB;
        iTermId = pEcbInfo->iTermId;
        iSyncId = pEcbInfo->iSyncId;
    }
}

```

```

if ( pEcbInfo && pEcbInfo->bFailed )
{
    bError == FALSE;
    return INT_CANCEL;
}

if ( oserr != DBNOERR )
{
    TMLog("DBLIB Error %s", oserrstr);
    if ( pEcbInfo )
    {
        pEcbInfo->bFailed = TRUE;
        bError = TRUE;
    }

    GetLocalTime(&systemTime);
    fp = fopen(szErrorLogPath, "ab");

    sprintf(szTmp, "ErrorHandler: DBLIB(%d): %s", oserr, oserrstr);

    TMLog("%2.2d/%2.2d/%2.2d %2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",
systemTime.wYear, systemTime.wMonth, systemTime.wDay,
systemTime.wSecond,
szTmp);

    fclose(fp);
}

return INT_CANCEL;
}

```

```

/* FUNCTION: int msg_handler(DBPROCESS *dbproc, DBINT msgno, int msgstate, int
severity, char *msgtext)
*
* PURPOSE:      This function handles DB-Library SQL Server error messages
*
* ARGUMENTS:   DBPROCESS      *dbproc          DBPROCESS id
pointer
*              message number    DBINT          msgno
*              msgstate          int           message state
*              severity          int           message severity
*              printable message description char *msgtext
*
* RETURNS:     int              INT_CONTINUE
*              continue if error is SQLETIME else INT_CANCEL action
*              INT_CANCEL          cancel operation
*
* COMMENTS:    This function also sets the dead lock dbproc variable if
necessary.
*/

int msg_handler(DBPROCESS *dbproc, DBINT msgno, int msgstate, int severity, char
*msgtext)
{
    PECBINFO
    pEcbInfo;

```

```

EXTENSION_CONTROL_BLOCK *pECB;
FILE *fp;
SYSTEMTIME systemTime;
char szTmp[256];
int iTermId;
int iSyncId;

if ( ! (pEcbInfo = (PECBINFO)dbgetuserdata(dbproc)) )
{
    pECB = gpECB;
    iTermId = 0;
    iSyncId = 0;
}
else
{
    pECB = pEcbInfo->pECB;
    iTermId = pEcbInfo->iTermId;
    iSyncId = pEcbInfo->iSyncId;
}

6006 ) if ( (msgno == 5701) || (msgno == 2528) || (msgno == 5703) || (msgno ==
        return INT_CONTINUE;

// deadlock message
if (msgno == 1205)
{
    // set the deadlock indicator
    if ( pEcbInfo )
        pEcbInfo->bDeadlock = TRUE;
    else
        TMLog("Error, dbgetuserdata returned NULL.");
    return INT_CONTINUE;
}
if ( pEcbInfo && pEcbInfo->bFailed )
{
    TMLog("SQL Error ");
    return INT_CANCEL;
}

if (msgno == 0)
    return INT_CONTINUE;
else
{
    TMLog("MsgHandler: SQL Error %s", msgtext);

    if ( pEcbInfo )
        pEcbInfo->bFailed = TRUE;

    bError = TRUE;

    GetLocalTime(&systemTime);

    sprintf(szTmp, "Error: QLSVR(%d): %s", msgno, msgtext);

    TMLog("%2.2d/%2.2d/%2.2d %2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",
        systemTime.wYear, systemTime.wMonth, systemTime.wDay,
        systemTime.wHour, systemTime.wMinute,
        systemTime.wSecond,
        szTmp);
    }
    return INT_CANCEL;

```

```

}

/* FUNCTION: BOOL GetParameters(int argc, char *argv[])
* PURPOSE: This function parses the command line passed in to the delivery
executable, initializing and filling in global variable parameters.
* ARGUMENTS: int argc number of command line arguments
passed to delivery char *argv[] array of command line
argument pointers
* RETURNS: BOOL FALSE parameter read successful
TRUE user has
requested parameter information screen be displayed.
* COMMENTS: None
*/

static BOOL GetParameters(int argc, char *argv[])
{
    int i;

    szServer[0] = 0;
    szPassword[0] = 0;
    bFlush = FALSE;
    strcpy(szDatabase, "tpcc");
    strcpy(szUser, "sa");

    for(i=0; i<argc; i++)
    {
        if ( argv[i][0] == '-' || argv[i][0] == '/' )
        {
            switch(argv[i][1])
            {
                case 'S':
                case 's':
                    strcpy(szServer, argv[i]+2);
                    break;
                case 'V':
                case 'v':
                    verbose = TRUE;
                    break;
                case '?':
                    return TRUE;
            }
        }
    }
    return FALSE;
}

/* FUNCTION: void PrintParameters(void)
* PURPOSE: This function displays the supported command line flags.
* ARGUMENTS: None
* RETURNS: None
* COMMENTS: None

```

```

*
*/

static void PrintParameters(void)
{
    TMLog("Performance Tuning Corporation Tuxedo Kit");
    TMLog(" www.perftuning.com (281) 251-3495 ");
    TMLog("NewOrder: -S Server -D Database -U User -P Password ");
    TMLog("NewOrder: Server %s, Database %s, User %s, Password %s.",
        szServer, szDatabase, szUser, szPassword);
}

```

resource.h

```

//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by TPCC.rc
//

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE        101
#define _APS_NEXT_COMMAND_VALUE        40001
#define _APS_NEXT_CONTROL_VALUE        1000
#define _APS_NEXT_SYMED_VALUE         101
#endif
#endif

```

testgv.c

```

#include "getval.h"
main()
{
    char s1[] = {"Test String DID*=08& CID*=123& IID*=54321& The end of test"};
    char s2[20];

    char * ptr;

    ptr = s1;

    GetVal(ptr, "DID*", s2, sizeof(s2))
    printf("Value of DID = %s\n",s2);

    GetVal(ptr, "CID*", s2, sizeof(s2))
    printf("Value of CID = %s\n",s2);

    GetVal(ptr, "IID*", s2, sizeof(s2))
    printf("Value of IID = %s\n",s2);
}

```

tpcc.c

```

/* FILE: TPCC.C
*
* Based on: Microsoft TPC-C Kit Ver. 3.00.000

```

```

*
* Copyright Microsoft, 1996
* Copyright Performance Tuning Corporation,
1997
*
* PURPOSE: TPC-C main program.
* Author: Philip Durr
*         philipdu@microsoft.com
*
* MODIFIED Changed for modularity and to allow for the Tuxedo TM
*
* Author: Edward Whalen
*         Performance Tuning Corporation
*         ewhalen@perftuning.com
*
*/

#define LOCAL_ALLOC 1

#define PERFORMIX
#define BOIL

#include <windows.h>
#include <process.h>
#include <stdio.h>
#include <stdarg.h>
#include <malloc.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <io.h>

#define DBNTWIN32
#include <sqlfront.h>
#include <sqlldb.h>

#include "trans.h" //tpckit
transaction header contains definations of structures specific to TPC-C
#include "httpext.h" //ISAPI DLL
information header
#include "getval.h"

#include "tpcc.h" //this dlls
specific structure, value e.t. header.

#include <tmenv.h>
#include <xa.h>
#include <atmi.h>

static TPINIT *tpinf;
static DWORD TLSIsTpInitedKey;
static DWORD TLSNewOrderKey;
static DWORD TLSPaymentKey;
static DWORD TLSOrderStatusKey;
static DWORD TLSDeliveryKey;
static DWORD TLSStockLevelKey;
static int ThrTpInit();

char szServer[32] = "EDW"; //global variables used with this
DLL
char szUser[32] = "sa";
char szPassword[32] = "";

```

```

char      szDatabase[32]      = "tpcc";
char      szIID[16][8]       =
{"IID00*", "IID01", "IID02", "IID03", "IID04", "IID05", "IID06", "IID07", "IID08", "IID09", "IID10", "IID11", "IID12", "IID13", "IID14", "IID15"};
char      szSP[16][8]        =
{"SP00*", "SP01", "SP02", "SP03", "SP04", "SP05", "SP06", "SP07", "SP08", "SP09", "SP10", "SP11", "SP12", "SP13", "SP14", "SP15"};
char      szQty[16][8]       =
{"Qty00*", "Qty01", "Qty02", "Qty03", "Qty04", "Qty05", "Qty06", "Qty07", "Qty08", "Qty09", "Qty10", "Qty11", "Qty12", "Qty13", "Qty14", "Qty15"};

BOOL      bLog                = FALSE;
BOOL      dLog                = FALSE;

int        iThreads            = 5;
int        iMaxWareHouses     = 0;
int        iDelayMs           = 100;
short      iDeadlockRetry     = (short)3;
short      iMaxConnections    = (short)25;
int        iErrVal             = 0;

char buffer[256];

//allowable client command strings i.e. CMD=command
char *szCmds[] =
{
    "..NewOrder..", "..Payment..", "..Delivery..", "..Order-Status..",
    "..Stock-Level..", "..Exit..",
    "Submit", "Begin", "Process", "Menu", "Clear", ""
};

//defined command string functions, called via CMD=command http string from html
client.

void (*DoCmd[]) (EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int
iSyncId) =
{
    NewOrderForm,
    PaymentForm,
    DeliveryForm,
    OrderStatusForm,
    StockLevelForm,
    ExitCmd,
    SubmitCmd,
    BeginCmd,
    ProcessCmd,
    MenuCmd,
    ClearCmd
};

//Terminal client id structure and interface definition
TERM      Term = { 0, 0, 0, FALSE, NULL, TermInit, TermAllocate, TermRestore,
TermAdd, TermDelete };

//welcome to tpc-c html form buffer, this is first form client sees.
static char *szWelcomeForm = "<HTML>"

    "<HEAD><TITLE>Welcome To TPC-C</TITLE></HEAD><BODY>"

    "Please Identify your Warehouse and District for this session.<BR>"

```

```

"<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
"<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">"
"<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">"
"<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"1\">"
"<INPUT TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"-2\">"
"<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"0\">"
"Warehouse ID <INPUT NAME=\"w_id\" SIZE=4><BR>"
"District ID <INPUT NAME=\"d_id\" SIZE=2><BR>"
"<HR>"
"<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Submit\">"
"</FORM><BODY>"
"</HTML>";

```

```

static char      szTpccLogPath[256]; //path to html log file if logging turned on
in registry.
static char szErrorLogPath[256];     //path to error log file.

static CRITICAL_SECTION      CriticalSection;
static CRITICAL_SECTION      ErrorLogCriticalSection;

static EXTENSION_CONTROL_BLOCK *gpECB;
static int                    bTpccExit;
//exit delivery disconnect loop as dll exiting.

/* FUNCTION: BOOL WINAPI DllMain(HANDLE hModule, DWORD ul_reason_for_call, LPVOID
lpReserved)
*
* PURPOSE:      This function is the entry point for the DLL this implementation
is based on the
*
*               fact that DLL_PROCESS_ATTACH is only called from the
inet service once. Connections
*
*               are sent to this function as thread attachments.
*
* ARGUMENTS:    HANDLE      hModule          module handle
*               DWORD       ul_reason_for_call reason for
call
*               LPVOID      lpReserved
*
*               reserved for future use
*
* RETURNS:      BOOL        FALSE
*               errors occured in initialization
*
*               DLL successfully initialized
*
* COMMENTS:     None
*/

BOOL WINAPI DllMain(HANDLE hModule, DWORD ul_reason_for_call, LPVOID lpReserved)
{

```



```

int i;
static SECURITY_ATTRIBUTES sa;
static PSECURITY_DESCRIPTOR pSD;

switch( ul_reason_for_call )
{
    case DLL_PROCESS_ATTACH:
        if ( ReadRegistrySettings() )
        {
            MessageBox(NULL, "Cannot Find TPCC Key in
registry (run install.exe).", "Init", MB_OK | MB_ICONSTOP);
            return FALSE;
        }

        InitializeCriticalSection(&CriticalSection);
        InitializeCriticalSection(&ErrorLogCriticalSection);

        (*Term.Init)();
        if ( !(*Term.Allocate)() )
        {
            MessageBox(NULL, "Error Trm.Allocate().",
"Init", MB_OK | MB_ICONSTOP);
            return FALSE;
        }
        for(i=Term.iNext; i<Term.iAvailable; i++)
            Term.pClientData[i].inUse = 0;
        Term.pClientData[0].inUse = 1;

        TLSIsTpInitedKey = TlsAlloc(); // check for failure

later
        TLSNewOrderKey = TlsAlloc();
        TLSPaymentKey = TlsAlloc();
        TLSOrderStatusKey = TlsAlloc();
        TLSDeliveryKey = TlsAlloc();
        TLSStockLevelKey = TlsAlloc();
        // assumption:value inited to 0
        break;
    case DLL_THREAD_ATTACH:
        break;
    case DLL_THREAD_DETACH:
        break;
    case DLL_PROCESS_DETACH:
        if ( pSD )
            free( pSD );

        bTpccExit = TRUE;

        (*Term.Restore)();

        DeleteCriticalSection(&CriticalSection);
        DeleteCriticalSection(&ErrorLogCriticalSection);

        TlsFree(TLSIsTpInitedKey);
        TlsFree(TLSNewOrderKey);
        TlsFree(TLSPaymentKey);
        TlsFree(TLSOrderStatusKey);
        TlsFree(TLSDeliveryKey);
        TlsFree(TLSStockLevelKey);

        break;
}
return TRUE;
}

```

```

/* FUNCTION: BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)
 *
 * PURPOSE: This function is called by the inet service when the DLL is
first loaded.
 *
 * ARGUMENTS: HSE_VERSION_INFO *pVer passed in structure in which to
place expected version number.
 *
 * RETURNS: TRUE inet service expected return value.
 *
 * COMMENTS: None
 */
BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVer)
{
    pVer->dwExtensionVersion = MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR);
    lstrcpyn(pVer->lpszExtensionDesc, "TPC-C Server.",
HSE_MAX_EXT_DLL_NAME_LEN);

    return TRUE;
}

/* FUNCTION: DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK *pECB)
 *
 * PURPOSE: This function is the main entry point for the TPCC DLL. The
internet service
calls this function passing in the http string.
 *
 * ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB structure pointer to
passed in internet
service information.
 *
 * RETURNS: DWORD HSE_STATUS_SUCCESS
connection can be dropped if error
HSE_STATUS_SUCCESS_AND_KEEP_CONN keep connect valid comment sent
 *
 * COMMENTS: None
 */
DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK *pECB)
{
    int iCmd, FormId, TermId, iSyncId;
    FILE *fp;

    static BOOL bReadRegistry = FALSE;

    if ( iMaxConnections == -1 )
    {
        ErrorMessage(pECB, ERR_CAN_NOT_SET_MAX_CONNECTIONS,
ERR_TYPE_WEBDLL, NULL, -1, -1);
        return HSE_STATUS_SUCCESS;
    }

    //if registry setting is for html logging then show http string passed in.
    if ( bLog )
    {

```

```

        SYSTEMTIME      systemTime;

        fp = fopen(szTpccLogPath, "ab");
        GetLocalTime(&systemTime);

        fprintf(fp, "** QUERY * %2.2d/%2.2d/%2.2d
%2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",
        systemTime.wYear, systemTime.wMonth, systemTime.wDay,
        systemTime.wHour, systemTime.wMinute,
        systemTime.wSecond,
        pECB->lpszQueryString);
        fclose(fp);
    }

    //process http query
    if ( !ProcessQueryString(pECB, &iCmd, &FormId, &TermId, &iSyncId ) )
    {
        if ( TermId < 0 )
            ErrorMessage(pECB, ERR_INVALID_TERMID,
ERR_TYPE_WEBDLL, NULL, TermId, iSyncId);
        else
            ErrorMessage(pECB, ERR_COMMAND_UNDEFINED,
ERR_TYPE_WEBDLL, NULL, TermId, iSyncId);
        return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
    }

    if ( TermId != 0 )
    {
        if ( !IsValidTermId(TermId) )
        {
            ErrorMessage(pECB, ERR_INVALID_TERMID,
ERR_TYPE_WEBDLL, NULL, TermId, iSyncId);
            return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
        }

        //must have a valid syncid here since termid is valid
        if ( iSyncId < 1 || iSyncId != Term.pClientData[TermId].iSyncId )
        {
            ErrorMessage(pECB, ERR_INVALID_SYNC_CONNECTION,
ERR_TYPE_WEBDLL, NULL, TermId, iSyncId);
            return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
        }
    }

    //set use time
    Term.pClientData[TermId].iTickCount = GetTickCount();

    //go execute http: command
    (*DoCmd[iCmd])(pECB, FormId, TermId, iSyncId);

    //finish up and keep connection
    return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

/* FUNCTION: static BOOL IsValidTermId(int TermId)
 *
 * PURPOSE:      This function checks to see of the passed in terminal id is
valid.
 *

```

```

 * ARGUMENTS:      int      TermId
                  client terminal id
 *
 * RETURNS:        BOOL      FALSE
                  Terminal ID Invalid
                  TRUE
                  Terminal ID valid
 *
 * COMMENTS:      None
 *
 */

static BOOL IsValidTermId(int TermId)
{
    return (BOOL) ( TermId > 0 && TermId <= Term.iAvailable &&
Term.pClientData[TermId].inUse );
}

/* FUNCTION: BOOL ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB, int *pCmd, int
*pFormId, int *pTermId, int *pSyncId)
 *
 * PURPOSE:      This function extracts the relevent information out of the http
command passed in from
 *
 *                the browser.
 *
 * ARGUMENTS:      EXTENSION_CONTROL_BLOCK *pECB      structure
                  pointer to passed in internet
 *
 *                service information.
 *
 *                int      returned command id
 *
 *                int      returned active form client browser is on
 *
 *                int      returned client terminal id
 *
 * RETURNS:        success      BOOL      FALSE
                  TRUE
                  command passed in is invalid
 *
 * COMMENTS:      If this is the initial connection i.e. client is at welcome
screen then
 *
 *                there will not be a terminal id or current
 *
 *                form id if this is the case
 *
 *                then the pTermId and pFormId return values
 *
 *                are undefined.
 *
 */

BOOL ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB, int *pCmd, int *pFormId, int
*pTermId, int *pSyncId)
{
    char *ptr;
    char szBuffer[25];
    char szTmp[25];
    char *dest = szBuffer;
    int i;

    if ( (ptr = strstr(pECB->lpszQueryString, "FORMID=")) )
        *pFormId = *(ptr+7) & 0x0F;

    if ( (ptr = strstr(pECB->lpszQueryString, "TERMID=")) )
    {

```

```

        *pTermId = atoi((ptr+7));
        if ( *pTermId == 0 )           //terminal id 0 used internally
            *pTermId = -1;
        if ( *pTermId == -2 )         //login screen
            *pTermId = 0;
    }
    else
        *pTermId = 0;

    if ( (ptr = strstr(pECB->lpszQueryString, "SYNCID=")) )
        *pSyncId = atoi((ptr+7));
    else
        *pSyncId = 0;

    if ( !(ptr = strstr(pECB->lpszQueryString, "CMD=")) )
    {
        ptr = szBuffer;
        if ( !strcmp(szBuffer, "Default") )
            strcpy(szBuffer, "CMD=Begin");
        switch( *pFormId )
        {
            case WELCOME_FORM:
                strcpy(szBuffer, "CMD=Submit");
                break;
            case MAIN_MENU_FORM:
                strcpy(szBuffer, "CMD=NewOrder");
                break;
            case NEW_ORDER_FORM:
            case PAYMENT_FORM:
            case DELIVERY_FORM:
            case ORDER_STATUS_FORM:
            case STOCK_LEVEL_FORM:
                if ( !(*pTermId) )
                    return FALSE;
                if ( GetKeyValue(pECB->lpszQueryString,
                    "PI*", szTmp, sizeof(szTmp)) )
                    strcpy(szBuffer, "CMD=Process");
                else
                {
                    strcpy(szBuffer, "CMD=");
                    strcat(szBuffer, szCmds[*pFormId -
NEW_ORDER_FORM]);
                }
                break;
            default:
                return FALSE;
        }
    }

    ptr += 4;

    while( *ptr && *ptr != '&' )
        *dest++ = *ptr++;
    *dest = 0;

    for(i=0; szCmds[i][0]; i++)
    {
        if ( !strcmp(szCmds[i], szBuffer) )
        {
            *pCmd = i;
            return TRUE;
        }
    }

```

```

    }
    return FALSE;
}

/* FUNCTION: void NewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
*
* PURPOSE:          This function wraps the functionality needed for the TPC-C New
Order Form.
*
* ARGUMENTS:       int          unused
                   iFormId      int
                   iTermId      id of calling browser, i.e. TERMID= from
http command line
*
                   EXTENSION_CONTROL_BLOCK *pECB
                   structure pointer to passed in internet
*
                   service information.
*
* RETURNS:         None
*
* COMMENTS:        None
*/

void NewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int
iSyncId)
{
    WriteZString(pECB, MakeNewOrderForm(iTermId, iSyncId, TRUE, FALSE));
    UNUSEDPARAM(iFormId);
    return;
}

/* FUNCTION: void PaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
*
* PURPOSE:          This function wraps the functionality needed for the TPC-C
Payment Form.
*
* ARGUMENTS:       int          unused
                   iFormId      int
                   iTermId      id of calling browser, i.e. TERMID= from
http command line
*
                   iSyncId      int
                   sync id of calling browser
                   EXTENSION_CONTROL_BLOCK *pECB
                   structure pointer to passed in internet
*
                   service information.
*
* RETURNS:         None
*
* COMMENTS:        None
*/

void PaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int
iSyncId)
{

```

```

        WriteZString(pECB, MakePaymentForm(iTermId, iSyncId, TRUE) );
        UNUSEDPARAM(iFormId);
    }

/* FUNCTION: void DeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
*
* PURPOSE:      This function wraps the functionality needed for the TPC-C
Delivery Form.
*
* ARGUMENTS:   int          unused
*              iFormId
*              iTermId      int
                        id of calling browser, i.e. TERMID= from
http command line
*              iSyncId     int
                        sync id of calling browser
*              EXTENSION_CONTROL_BLOCK *pECB
                        structure pointer to passed in internet
*
* RETURNS:     None
*
* COMMENTS:    None
*/

void DeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int
iSyncId)
{
    WriteZString(pECB, MakeDeliveryForm(iTermId, iSyncId, TRUE) );
    UNUSEDPARAM(iFormId);
}

/* FUNCTION: void OrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
*
* PURPOSE:      This function wraps the functionality needed for the TPC-C Order
Status Form.
*
* ARGUMENTS:   int          unused
*              iFormId
*              iTermId      int
                        id of calling browser, i.e. TERMID= from
http command line
*              iSyncId     int
                        sync id of calling browser
*              EXTENSION_CONTROL_BLOCK *pECB
                        structure pointer to passed in internet
*
* RETURNS:     None
*
* COMMENTS:    None
*/

void OrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int
iSyncId)
{

```

```

        WriteZString(pECB, MakeOrderStatusForm(iTermId, iSyncId, TRUE) );
        UNUSEDPARAM(iFormId);
    }

/* FUNCTION: void StockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
*
* PURPOSE:      This function wraps the functionality needed for the TPC-C Stock
Level Form.
*
* ARGUMENTS:   int          unused
*              iFormId
*              iTermId      int
                        id of calling browser, i.e. TERMID= from
http command line
*              iSyncId     int
                        sync id of calling browser
*              EXTENSION_CONTROL_BLOCK *pECB
                        structure pointer to passed in internet
*
* RETURNS:     None
*
* COMMENTS:    None
*/

void StockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int
iSyncId)
{
    WriteZString(pECB, MakeStockLevelForm(iTermId, iSyncId, TRUE) );
    return;
}

/* FUNCTION: void Exitcmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
*
* PURPOSE:      This function removes a terminal id from use, the allocated
structure however remains
valid so the next request for a new client will not
require a new memory allocation.
*
* ARGUMENTS:   int          unused
*              iFormId
*              iTermId      int
                        id of calling browser, i.e. TERMID= from
http command line
*              iSyncId     int
                        sync id of calling browser
*              EXTENSION_CONTROL_BLOCK *pECB
                        structure pointer to passed in internet
*
* RETURNS:     None
*
* COMMENTS:    None
*/

void Exitcmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int iSyncId)

```

```

{
    (*Term.Delete)(pECB, iTermId);

    WriteZString(pECB, MakeWelcomeForm() );

    UNUSEDPARAM(iFormId);
    UNUSEDPARAM(iSyncId);

    return;
}

/* FUNCTION: void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
*
* PURPOSE:          This function allocated a new terminal id in the Term structure
array.
*
* ARGUMENTS:       int          unused
                   iFormId      unused
                   iTermId      int          id of calling browser, i.e. TERMIID= from
http command line
*
                   iSyncId      int          sync id of calling browser
                   EXTENSION_CONTROL_BLOCK *pECB
                   structure pointer to passed in internet
*
* RETURNS:         None          service information.
*
* COMMENTS:        A terminal id can be allocated but still be invalid if the
requested warehouse number
*
                   is outside the range specified in the
registry. This then will force the client id
*
                   to be invalid and an error message sent to
the users browser.
*/

void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int iSyncId)
{
    int          iCurrent;

    if ( iCurrent = (*Term.Add)(pECB, pECB->lpszQueryString) < 0 )
    {
        ErrorMessage(pECB, ERR_CANNOT_INIT_TERMINAL, ERR_TYPE_WEBDLL,
NULL, iCurrent, iSyncId);
        return;
    }

    if ( Term.pClientData[iCurrent].w_id > iMaxWareHouses ||
Term.pClientData[iCurrent].w_id < 1 )
    {
        ErrorMessage(pECB, ERR_W_ID_INVALID, ERR_TYPE_WEBDLL, NULL,
iCurrent, iSyncId);
        (*Term.Delete)(pECB, iCurrent);
        return;
    }

    if ( Term.pClientData[iCurrent].d_id < 1 ||
Term.pClientData[iCurrent].d_id > 10 )
    {
        ErrorMessage(pECB, ERR_D_ID_INVALID, ERR_TYPE_WEBDLL, NULL,
iCurrent, iSyncId);
        (*Term.Delete)(pECB, iCurrent);
    }
}

```

```

        return;
    }

    WriteZString(pECB, MakeMainMenuForm(iCurrent,
Term.pClientData[iCurrent].iSyncId) );

    return;
}

/* FUNCTION: void BeginCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
*
* PURPOSE:          This function is the first command executed. It is executed with
the command
                   CMD=Begin?Server=xxx from the http command line.
*
* ARGUMENTS:       int          unused
                   iFormId      unused
                   iTermId      int          id of calling browser, i.e. TERMIID= from
http command line
*
                   iSyncId      int          sync id of calling browser
                   EXTENSION_CONTROL_BLOCK *pECB
                   structure pointer to passed in internet
*
* RETURNS:         None          service information.
*
* COMMENTS:        SQL server must be specified, however the user and password
parameters are optional.
*
                   The complete command line is
CMD=Begin&Server=server&User=sa&Psw=&. The & are used
*
                   to separate parameters which is internet
browser standard.
*/

void BeginCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int iSyncId)
{
    LPSTR pQueryString;

    pQueryString = pECB->lpszQueryString;

    /*
    if ( !GetKeyValue(pQueryString, "Server", szServer, sizeof(szServer)) )
    {
        ErrorMessage(pECB, ERR_NO_SERVER_SPECIFIED, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
        return;
        strcpy(szServer, "TPCC");
    }

    if ( !GetKeyValue(pQueryString, "User", szUser, sizeof(szUser)) )
        strcpy(szUser, "sa");

    if ( !GetKeyValue(pQueryString, "Psw", szPassword, sizeof(szPassword)) )
        strcpy(szPassword, "");

    if ( !GetKeyValue(pQueryString, "Db", szDatabase, sizeof(szDatabase)) )
        strcpy(szDatabase, "tpcc");
    */

    WriteZString(pECB, MakeWelcomeForm() );

    UNUSEDPARAM(iFormId);
}

```

```

        return;
    }

/* FUNCTION: void ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int
iTermId, int iSyncId)
*
* PURPOSE:      This function process the passed in http command
*
* ARGUMENTS:   int          unused
*              iFormId      int
*              iTermId      id of calling browser, i.e. TERMID= from
http command line
*
*              iSyncId      int
*              sync id of calling browser
*              EXTENSION_CONTROL_BLOCK *pECB
*              structure pointer to passed in internet
*
* RETURNS:     None          service information.
*
* COMMENTS:    None
*
*/

void ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int
iSyncId)
{
    switch( iFormId )
    {
        case WELCOME_FORM:
            return;
        case MAIN_MENU_FORM:
            return;
        case NEW_ORDER_FORM:
            ProcessNewOrderForm(pECB, iTermId, iSyncId);
            return;
        case PAYMENT_FORM:
            ProcessPaymentForm(pECB, iTermId, iSyncId);
            return;
        case DELIVERY_FORM:
            ProcessDeliveryForm(pECB, iTermId, iSyncId);
            return;
        case ORDER_STATUS_FORM:
            ProcessOrderStatusForm(pECB, iTermId, iSyncId);
            return;
        case STOCK_LEVEL_FORM:
            ProcessStockLevelForm(pECB, iTermId, iSyncId);
            return;
    }
}

/* FUNCTION: void ClearCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
*
* PURPOSE:      This function frees all currently logged in terminal ids.
*
* ARGUMENTS:   int          unused
*              iFormId      int
*              iTermId      id of calling browser, i.e. TERMID= from
http command line

```

```

*              int
*              sync id of calling browser
*              EXTENSION_CONTROL_BLOCK *pECB
*              structure pointer to passed in internet
*
* RETURNS:     None          service information.
*
* COMMENTS:    Use this function with caution, it may cause unpredictable
results
*              if existing browsers attempt to use the web
client with out
*              beginning at the login screen for each
client.
*/

void ClearCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int iSyncId)
{
    int i;

    EnterCriticalSection(&CriticalSection);

    for(i=0; i<Term.iAvailable; i++)
    {
        if ( Term.pClientData[i].inUse )
            (*Term.Delete)(pECB, i);
    }

    Term.iNext = 0;
    Term.iAvailable = 0;
    Term.iMasterSyncId = 1;

    if ( Term.pClientData )
        free(Term.pClientData);
    Term.pClientData = NULL;
    Term.bInit = FALSE;

    (*Term.Init)();
    if ( !(*Term.Allocate)() )
    {
        ErrorMessage(pECB, ERR_MAX_CONNECT_PARAM, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }
    for(i=Term.iNext; i<Term.iAvailable; i++)
        Term.pClientData[i].inUse = 0;
    Term.pClientData[0].inUse = 1;

    LeaveCriticalSection(&CriticalSection);

    WriteZString(pECB, MakeWelcomeForm() );

    return;
}

/* FUNCTION: void MenuCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId,
int iSyncId)
*
* PURPOSE:      This function causes an exit to the main menu
*
* ARGUMENTS:   int          unused
*              iFormId      unused

```

```

*
*          iTermId          int
http command line          id of calling browser, i.e. TERMIID= from
*
*          iSyncId          int
*                          sync id of calling browser
*                          EXTENSION_CONTROL_BLOCK *pECB
*          structure pointer to passed in internet
*
*          service information.
* RETURNS:          None
*
* COMMENTS:          None
*/

void MenuCmd(EXTENSION_CONTROL_BLOCK *pECB, int iFormId, int iTermId, int iSyncId)
{
    WriteZString(pECB, MakeMainMenuForm(iTermId, iSyncId) );
    return;
}

/* FUNCTION: void WriteZString(EXTENSION_CONTROL_BLOCK *pECB, char *szStr)
*
* PURPOSE:          This function is the low level output function. It writes a
string of text back to the
*                  client browser.
*
* ARGUMENTS:          EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*                  char *szStr
*                  string to display in the client browser.
*
* RETURNS:          None
*
* COMMENTS:          This function assumes that the string to written to the client
browser has
*                  been formatted in an HTML manner.
*/

static void WriteZString(EXTENSION_CONTROL_BLOCK *pECB, char *szStr)
{
    FILE *fp;
    int lpbSize;
#ifdef PERFORMIX
    int iSize;
    char szHeader[128];
    char szHeader1[128];
#endif

    lpbSize = strlen(szStr)+1;

    if ( bLog )
    {
        SYSTEMTIME systemTime;

        fp = fopen(szTpcLogPath, "ab");

        GetLocalTime(&systemTime);

        fprintf(fp, "** HTML PAGE * %2.2d/%2.2d/%2.2d
%2.2d:%2.2d:%2.2d\r\n\r\n%s\r\n\r\n",

```

```

        systemTime.wYear, systemTime.wMonth, systemTime.wDay,
        systemTime.wSecond,
        systemTime.wHour, systemTime.wMinute,
        szStr);
    }
    fclose(fp);
}

#ifdef PERFORMIX
    (*pECB->ServerSupportFunction)(pECB->ConnID, HSE_REQ_DONE_WITH_SESSION,
NULL, 0, 0);

    (*pECB->WriteClient)(pECB->ConnID, szStr, &lpbSize, 0);
#else
    iSize = sprintf(szHeader, "200 Ok");
    sprintf(szHeader1, "Connection: keep-alive\r\nContent-type:
text/html\r\nContent-length: %d\r\n\r\n", lpbSize);
    (*pECB->ServerSupportFunction)(pECB->ConnID, HSE_REQ_SEND_RESPONSE_HEADER,
szHeader, &iSize, (LPDWORD)szHeader1);

    (*pECB->WriteClient)(pECB->ConnID, szStr, &lpbSize, 0);
#endif

    return;
}

/* FUNCTION: void h_printf(EXTENSION_CONTROL_BLOCK *pECB, char *format, ...)
*
* PURPOSE:          This function forms a high level printf for an HTML browser
*
* ARGUMENTS:          EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*                  char *format
*                  printf style format string
*                  ...
*                  other arguments as required by printf style
format string.
*
* RETURNS:          None
*
* COMMENTS:          This function is mainly used for developmental support.
*/

static void h_printf(EXTENSION_CONTROL_BLOCK *pECB, char *format, ...)
{
    int lpbSize;
    char szBuff[512];
    char szTmp[512];

    va_list marker;
    va_start( marker, format );
    vsprintf(szTmp, format, marker);
    va_end( marker );

    lpbSize = wsprintf(szBuff, "<html>%s</html>", szTmp) + 1;

    (*pECB->WriteClient)(pECB->ConnID, szBuff, &lpbSize, 0);

    return;
}

```

```

/* FUNCTION: void ErrorMessage(EXTENSION_CONTROL_BLOCK *pECB, int iError, int
iErrorType, char *szMsg)
*
* PURPOSE:      This function displays an error message in the client browser.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*
*              int
*              id of error message
*              int
*              error type, ERR_TYPE_SQL, ERR_TYPE_DBLIB, or
ERR_TYPE_WEBDLL
*              int
*              terminal id from browser
*              int
*              sync id from browser
*              char *
*              szMsg
*              optional error message string used with ERR_TYPE_SQL and
ERR_TYPE_DBLIB
*
* RETURNS:     None
*
* COMMENTS:    If the error type is ERR_TYPE_WEBDLL the szmsg parameter may be
NULL because it
*              is ignored. If the error type is
ERR_TYPE_SQL or ERR_TYPE_DBLIB then the szMsg
*              parameter contains the text of the error
message, so the szMsg parameter cannot
*              be NULL.
*/

```

```

void ErrorMessage(EXTENSION_CONTROL_BLOCK *pECB, int iError, int iErrorType, char
*szMsg, int iTermId, int iSyncId)
{
    int i;

    static SERRORMSG errorMsgs[] =
    {
        { ERR_SUCCESS,
          "Success, no error."
        },
        { ERR_COMMAND_UNDEFINED,
          "Command undefined."
        },
        { ERR_NOT_IMPLEMENTED_YET,
          "Not Implemented Yet."
        },
        { ERR_CANNOT_INIT_TERMINAL,
          "Cannot initialize client connection."
        },
        { ERR_OUT_OF_MEMORY,
          "insufficient memory."
        },
        { ERR_NEW_ORDER_NOT_PROCESSED,
          "Cannot process new Order form."
        },
        { ERR_PAYMENT_NOT_PROCESSED,
          "Cannot process payment form."
        }
    },

```

```

        { ERR_NO_SERVER_SPECIFIED,
          "No Server name specified."
        },
        { ERR_ORDER_STATUS_NOT_PROCESSED,
          "Cannot process order status form."
        },
        { ERR_W_ID_INVALID,
          "Invalid Warehouse ID."
        },
        { ERR_CAN_NOT_SET_MAX_CONNECTIONS,
          "Insufficient memory to allocate # Connections."
        },
        { ERR_NOSUCH_CUSTOMER,
          "No such customer."
        },
        { ERR_D_ID_INVALID,
          "Invalid District ID Must be 1 to 10."
        },
        { ERR_MAX_CONNECT_PARAM,
          "Max client connections exceeded, run install to increase."
        },
        { ERR_INVALID_SYNC_CONNECTION,
          "Invalid Terminal Sync ID."
        },
        { ERR_INVALID_TERMID,
          "Invalid Terminal ID."
        },
        { ERR_PAYMENT_INVALID_CUSTOMER,
          "Payment Form, No such Customer."
        },
        { ERR_SQL_OPEN_CONNECTION,
          "SQLOpenConnection API Failed."
        },
        { ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
          "Stock Level
missing Threshold key \"TT*\"."
        },
        { ERR_STOCKLEVEL_THRESHOLD_INVALID,
          "Stock Level Threshold invalid data type range = 1 - 99."
        },
        { ERR_STOCKLEVEL_THRESHOLD_RANGE,
          "Stock Level Threshold Out of range, range must be 1 - 99."
        },
        { ERR_STOCKLEVEL_NOT_PROCESSED,
          "Stock Level not processed."
        },
        { ERR_NEWORDER_FORM_MISSING_DID,
          "New Order missing District key \"DID*\"."
        },
        { ERR_NEWORDER_DISTRICT_INVALID,
          "New Order District ID Invalid range 1 - 10."
        },
        { ERR_NEWORDER_DISTRICT_RANGE,
          "New
Order District ID out of Range. Range = 1 - 10."
        },
        { ERR_NEWORDER_CUSTOMER_KEY,
          "New Order missing Customer key \"CID*\"."
        },
        { ERR_NEWORDER_CUSTOMER_INVALID,
          "New Order customer id invalid data type, range = 1 to 3000."
        },
        { ERR_NEWORDER_CUSTOMER_RANGE,
          "New
Order customer id out of range, range = 1 to 3000."
        }
    },

```



```

    {
        ERR_NEWORDER_MISSING_IID_KEY,
    "New
Order missing Item Id key \"IID*\"."
    },
    {
        ERR_NEWORDER_ITEM_BLANK_LINES,
        "New Order blank order lines all orders must be continuous."
    },
    {
        ERR_NEWORDER_ITEMID_INVALID,
    "New
Order Item Id is wrong data type, must be numeric."
    },
    {
        ERR_NEWORDER_MISSING_SUPPW_KEY,
        "New Order missing Supp_W key \"SP##*\"."
    },
    {
        ERR_NEWORDER_SUPPW_INVALID,
        "New Order Supp_W invalid data type must be numeric."
    },
    {
        ERR_NEWORDER_MISSING_QTY_KEY,
    "New
Order Missing Qty key \"Qty##*\"."
    },
    {
        ERR_NEWORDER_QTY_INVALID,
        "New Order Qty invalid must be numeric range 1 - 99."
    },
    {
        ERR_NEWORDER_SUPPW_RANGE,
        "New Order Supp_W value out of range range = 1 - Max Warehouses."
    },
    {
        ERR_NEWORDER_ITEMID_RANGE,
        "New Order Item Id is out of range. Range = 1 to 999999."
    },
    {
        ERR_NEWORDER_QTY_RANGE,
        "New Order Qty is out of range. Range = 1 to 99."
    },
    {
        ERR_PAYMENT_DISTRICT_INVALID,
        "Payment District ID is invalid must be 1 - 10."
    },
    {
        ERR_NEWORDER_SUPPW_WITHOUT_ITEMID,
    "New
Order Supp_W field entered without a corresponding Item Id."
    },
    {
        ERR_NEWORDER_QTY_WITHOUT_ITEMID,
    "New
Order Qty entered without a corresponding Item Id."
    },
    {
        ERR_NEWORDER_NOITEMS_ENTERED,
    "New
Order Blank Items between items, items must be continuous."
    },
    {
        ERR_PAYMENT_MISSING_DID_KEY,
        "Payment missing District Key \"DID*\"."
    },
    {
        ERR_PAYMENT_DISTRICT_RANGE,
        "Payment District Out of range, range = 1 - 10."
    },
    {
        ERR_PAYMENT_MISSING_CID_KEY,
        "Payment missing Customer Key \"CID*\"."
    },
    {
        ERR_PAYMENT_CUSTOMER_INVALID,
        "Payment Customer data type invalid, must be numeric."
    },
    {
        ERR_PAYMENT_MISSING_CLT,
        "Payment missing Customer Last Name Key \"CLT*\"."
    },
    {
        ERR_PAYMENT_LAST_NAME_TOO_LONG,
        "Payment Customer last name longer than 16 characters."
    },
    {
        ERR_PAYMENT_CUSTOMER_RANGE,
        "Payment Customer ID out of range, must be 1 to 3000."
    },
    {
        ERR_PAYMENT_CID_AND_CLT,
        "Payment Customer ID and Last Name entered must be one or other."
    },
    {
        ERR_PAYMENT_MISSING_CDI_KEY,
        "Payment missing Customer district key \"CDI*\"."
    },

```

```

    {
        ERR_PAYMENT_CDI_INVALID,
    "Payment Customer district invalid must be numeric."
    },
    {
        ERR_PAYMENT_CDI_RANGE,
        "Payment Customer district out of range must be 1 - 10."
    },
    {
        ERR_PAYMENT_MISSING_CWI_KEY,
    "Payment missing Customer Warehouse key \"CWI*\"."
    },
    {
        ERR_PAYMENT_CWI_INVALID,
    "Payment Customer Warehouse invalid must be numeric."
    },
    {
        ERR_PAYMENT_CWI_RANGE,
        "Payment Customer Warehouse out of range, 1 to Max Warehouses."
    },
    {
        ERR_PAYMENT_MISSING_HAM_KEY,
    "Payment missing Amount key \"HAM*\"."
    },
    {
        ERR_PAYMENT_HAM_INVALID,
    "Payment Amount invalid data type must be numeric."
    },
    {
        ERR_PAYMENT_HAM_RANGE,
        "Payment Amount out of range, 0 - 9999.99."
    },
    {
        ERR_ORDERSTATUS_MISSING_DID_KEY,
    "Order Status missing District key \"DID*\"."
    },
    {
        ERR_ORDERSTATUS_DID_INVALID,
    "Order Status District invalid, value must be numeric 1 - 10."
    },
    {
        ERR_ORDERSTATUS_DID_RANGE,
    "Order Status District out of range must be 1 - 10."
    },
    {
        ERR_ORDERSTATUS_MISSING_CID_KEY,
    "Order Status missing Customer key \"CID*\"."
    },
    {
        ERR_ORDERSTATUS_MISSING_CLT_KEY,
    "Order Status missing Customer Last Name key \"CLT*\"."
    },
    {
        ERR_ORDERSTATUS_CLT_RANGE,
    "Order Status Customer last name longer than 16 characters."
    },
    {
        ERR_ORDERSTATUS_CID_INVALID,
    "Order Status Customer ID invalid, range must be numeric 1 - 3000."
    },
    {
        ERR_ORDERSTATUS_CID_RANGE,
    "Order Status Customer ID out of range must be 1 - 3000."
    },
    {
        ERR_ORDERSTATUS_CID_AND_CLT,
    "Order Status Customer ID and LastName entered must be only one."
    },
    {
        ERR_DELIVERY_MISSING_OCD_KEY,
    "Delivery missing Carrier ID key \"OCD*\"."
    },
    {
        ERR_DELIVERY_CARRIER_INVALID,
    "Delivery Carrier ID invalid must be numeric 1 - 10."
    },
    {
        ERR_DELIVERY_CARRIER_ID_RANGE,
    "Delivery Carrier ID out of range must be 1 - 10."
    },
    {
        ERR_PAYMENT_MISSING_CLT_KEY,
    "Payment missing Customer Last Name key \"CLT*\"."
    },
    {
        ERR_TPINIT_BAD,
        "Tuxedo tpinit Failed."
    },

```

```

    },
    {
        ERR_TPALLOC_BAD,
        "Tuxedo tpalloc Failed."
    },
    {
        ERR_TPCALL_BAD,
        "Tuxedo tpcall Failed."
    },
    {
        0,
        ""
    }
};

static char szNoMsg[] = "";
char *szForm;

if ( !szMsg )
    szMsg = szNoMsg;

if ( iTermId > 0 && IsValidTermId(iTermId) )
    szForm = Term.pClientData[iTermId].szBuffer; //if termid valid
use common terminal static buffer.
else
    szForm = Term.pClientData[0].szBuffer; //else term id invalid
so use common terminal static buffer.
switch(iErrorType)
{
    case ERR_TYPE_WEBDLL:
        for(i=0; errorMsgs[i].szMsg[0]; i++)
        {
            if ( iError == errorMsgs[i].iError )
                break;
        }
        if ( !errorMsgs[i].szMsg[0] )
            i = 1;
        strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To
TPC-C</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">");
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">", iError);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
        wsprintf(szForm+strlen(szForm), "Error: TPCCWEB(%d):
%s", iError, errorMsgs[i].szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;
    case ERR_TYPE_SQL:
        strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To
TPC-C</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">");
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">", iError);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);

```

```

        wsprintf(szForm+strlen(szForm), "Error: SQLSVR(%d):
%s", iError, szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;
    case ERR_TYPE_DBLIB:
        strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To
TPC-C</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">");
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">", iError);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
        wsprintf(szForm+strlen(szForm), "Error: DBLIB(%d):
%s", iError, szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;
    case ERR_TYPE_TUXEDO:
        strcpy(szForm, "<HTML><HEAD><TITLE>Welcome To
TPC-C</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">");
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">", iError);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"%d\">", iErrVal);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"TERMID\" VALUE=\"%d\">", iTermId);
        wsprintf(szForm+strlen(szForm), "<INPUT
TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iSyncId);
        wsprintf(szForm+strlen(szForm), "Error: Tuxedo: %d
%s", iError, szMsg);
        strcat(szForm, "</FORM><BODY></HTML>");
        WriteZString(pECB, szForm);
        break;
}
return;
}

/* FUNCTION: BOOL GetKeyValue(char *pQueryString, char *pKey, char *pValue, int
iMax)
*
* PURPOSE: This function parses a http formatted string for specific key
values.
*
* ARGUMENTS: char *pQueryString http string
from client browser
* char *pKey
key value to look for
* char *pValue
character array into which to place key's value
* int iMax
maximum length of key value array.
* RETURNS: BOOL FALSE key value not found
TRUE key valud
found
*
* COMMENTS: http keys are formatted either KEY=value& or KEY=value\0. This
DLL formats

```

```

*
*           TPC-C input fields in such a manner that the
keys can be extracted in the
*           above manner.
*/

static BOOL GetKeyValue(char *pQueryString, char *pKey, char *pValue, int iMax)
{
    char *ptr;

    if ( !(ptr=strstr(pQueryString, pKey)) )
        return FALSE;
    if ( !(ptr=strchr(ptr, '=') ) )
        return FALSE;
    ptr++;
    iMax--;
    while( *ptr && *ptr != '&' && iMax)
    {
        *pValue++ = *ptr++;
        iMax--;
    }
    *pValue = 0;
    return TRUE;
}

/* FUNCTION: void TermInit(void)
*
* PURPOSE:      This function initializes the client terminal structure it is
called when the TPCC.DLL
*               is first loaded by the inet service.
*
* ARGUMENTS:    none
*
* RETURNS:      None
*
* COMMENTS:     None
*/

static void TermInit(void)
{
    if ( Term.bInit )
        return;
    Term.iNext
    Term.iMasterSyncId = 1;

    Term.iAvailable
    Term.pClientData = NULL;
    Term.bInit
    = TRUE;

    return;
}

/* FUNCTION: void TermRestore(void)
*
* PURPOSE:      This function frees allocated resources associated with the
terminal structure.
*
* ARGUMENTS:    none
*
* RETURNS:      None
*
*/

```

```

* COMMENTS:     This function is called only with the inet service unloads the
TPCC.DLL
*
*/

static void TermRestore(void)
{
    Term.iNext
    Term.iAvailable
    Term.iMasterSyncId = 0;
    if ( Term.pClientData )
        free(Term.pClientData);
    Term.pClientData = NULL;
    Term.bInit
    = FALSE;

    return;
}

/* FUNCTION: int TermAllocate(void)
*
* PURPOSE:      This function allocates more terminal array entries in the Term
structure.
*
* ARGUMENTS:    None
*
* RETURNS:      int TRUE or 1 if successfull
                int FALSE or 0 if terminal id cannot
be allocated.
*
* COMMENTS:     None
*/

static int TermAllocate(void)
{
    Term.iAvailable += 32;
    if ( !Term.pClientData )
        Term.pClientData = (PCLIENTDATA)malloc(Term.iAvailable *
sizeof(CLIENTDATA));
    else
        Term.pClientData = (PCLIENTDATA)realloc(Term.pClientData,
Term.iAvailable * sizeof(CLIENTDATA));
    return ( Term.pClientData ) ? 1 : 0;
}

/* FUNCTION: int TermAdd(EXTENSION_CONTROL_BLOCK *pECB, char *pQueryString)
*
* PURPOSE:      This function assigns a terminal id which is used to identify a
client browser.
*
* ARGUMENTS:    EXTENSION_CONTROL_BLOCK *pECB
                passed in structure pointer from inetsrv.
                char
                *pQueryString http query string passed to this DLL.
*
* RETURNS:      int assigned terminal id
                -1 cannot assign id error
occured.
*
* COMMENTS:     if the terminal id cannot be assigned it is because of
insufficient memory or the

```

```

*                               SQL connection cannot be allocated.
*
*/

static int TermAdd(EXTENSION_CONTROL_BLOCK *pECB, char *pQueryString)
{
    char    szTmp[32];
    int     i, iCurrent, iTotConnections, iTickCount;

    EnterCriticalSection(&CriticalSection);

    for(i=0, iTotConnections = 0; i<Term.iAvailable; i++)
    {
        if ( Term.pClientData[i].inUse )
            iTotConnections++;
    }

    if ( iTotConnections >= iMaxConnections )
    {
        for(iCurrent = 1, i=1, iTickCount = 0x7FFFFFFF;
i<iMaxConnections; i++)
        {
            if ( iTickCount > Term.pClientData[i].iTickCount )
            {
                iTickCount = Term.pClientData[i].iTickCount;
                iCurrent = i;
            }
        }
    }
    else
    {
        for(i=0; i<Term.iAvailable; i++)
        {
            if ( !Term.pClientData[i].inUse )
                break;
        }
        iCurrent = i;
    }

    if ( i == Term.iAvailable )
    {
        Term.iNext = Term.iAvailable;
        if ( !(*Term.Allocate)() )
            goto TermAddErr1;
        for(i=Term.iNext; i<Term.iAvailable; i++)
            Term.pClientData[i].inUse = 0;
        iCurrent = Term.iNext;
    }

    Term.pClientData[iCurrent].inUse = 1;

    if ( !GetKeyValue(pQueryString, "w_id", szTmp, sizeof(szTmp)) )
        goto TermAddErr1;

    Term.pClientData[iCurrent].w_id = (short)atoi(szTmp);

    if ( !GetKeyValue(pQueryString, "d_id", szTmp, sizeof(szTmp)) )
        goto TermAddErr1;

    Term.pClientData[iCurrent].d_id = atoi(szTmp);

    Term.pClientData[iCurrent].iTickCount = GetTickCount();
    Term.pClientData[iCurrent].iSyncId = Term.iMasterSyncId++;
}

```

```

        if ( Init(pECB, iCurrent, Term.pClientData[iCurrent].iSyncId, szServer,
szUser, szPassword, szDatabase) )
        {
            (*Term.Delete)(pECB, iCurrent);
            goto TermAddErr1;
        }

        LeaveCriticalSection(&CriticalSection);
        return iCurrent;

TermAddErr1:
    LeaveCriticalSection(&CriticalSection);
    return -1; //terminal unsuccessfully added
}

/* FUNCTION: void TermDelete(EXTENSION_CONTROL_BLOCK *pECB, int id)
*
* PURPOSE:      This function makes a terminal entry in the Term array available
for reuse.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*
*              int
*              id Terminal id of client exiting
*
* RETURNS:     None
*
* COMMENTS:    None
*
*/

static void TermDelete(EXTENSION_CONTROL_BLOCK *pECB, int id)
{
    if ( id >= 0 && id < Term.iAvailable )
    {
        Close(pECB, id, -1);
        Term.pClientData[id].inUse = 0;
#ifdef LOCAL_ALLOC
        tpfree((char *)Term.pClientData[id].NewOrderDataPtr);
        tpfree((char *)Term.pClientData[id].PaymentDataPtr);
        tpfree((char *)Term.pClientData[id].OrderStatusDataPtr);
        tpfree((char *)Term.pClientData[id].DeliveryDataPtr);
        tpfree((char *)Term.pClientData[id].StockLevelDataPtr);
#endif
    }

    return;
}

/* FUNCTION: BOOL Init(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId, char
*szServer, char *szUser, char *szPassword, char *szDatabase)
*
* PURPOSE:      This function initializes the sql connection for use.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB passed in
structure pointer from inetsrv.
*
*              int
*              iTermId id of browser client that this connection is
for.
*              int
*              iSyncId sync id for this client session

```

```

*          char
*          *szServer sql server name
*          char
*          *szUser      user name
*          char
*          *szPassword  user password
*          char
*          *szDatabase  database to use
*
* RETURNS:      BOOL      FALSE      if successfull
*              TRUE      if an error
occurs and connection cannot be established.
*
* COMMENTS:      None
*
*/

BOOL Init(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId, char *szServer,
char *szUser, char *szPassword, char *szDatabase)
{
    char      szApp[32];
#ifdef LOCAL_ALLOC
    char      buf[40];
    int       iRc;
#endif

    sprintf(szApp, "TPCC:%ld", (int)iTermId);
    Term.pClientData[iTermId].dbproc = NULL;

    /*      if ( SQLOpenConnection(pECB, iTermId, iSyncId,
&Term.pClientData[iTermId].dbproc, szServer, szDatabase, szUser, szPassword, szApp,
&Term.pClientData[iTermId].spid) )
    {
        ErrorMessage(pECB, ERR_SQL_OPEN_CONNECTION, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
        return TRUE;
    }
*/

#ifdef LOCAL_ALLOC
//      Add initialization of Tuxedo Structures

//          NEWORDER

    if ((Term.pClientData[iTermId].NewOrderDataPtr = (NEW_ORDER_DATA
*)tpalloc("CARRAY", NULL, sizeof(NEW_ORDER_DATA))) == NULL)
    {
        iRc = tperrno;
        sprintf(buf, "Tpcalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO,
buf, 0, 0);

        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> Init Thread %d : NewOrder

tpalloc failed: iRc = %d \r\n",

                GetCurrentThreadId(), iRc);
            fclose(fp);
        }
        sprintf(buf, "Tpcalloc %d", iRc);

```

```

        ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO,
buf, 0, 0);

        return TRUE;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** Init Thread %d iTermId %d *

NewOrderDataPtr: %x \r\n",

                GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].NewOrderDataPtr);
        fclose(fp);
    }

//          PAYMENT

    if ((Term.pClientData[iTermId].PaymentDataPtr = (PAYMENT_DATA
*)tpalloc("CARRAY", NULL, sizeof(PAYMENT_DATA))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> Init Thread %d : Payment

tpalloc failed: iRc = %d \r\n",

                GetCurrentThreadId(), iRc);
            fclose(fp);
        }
        sprintf(buf, "Tpcalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO,
buf, 0, 0);

        return TRUE;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** Init Thread %d iTermId %d *

PaymentDataPtr: %x \r\n",

                GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].PaymentDataPtr);
        fclose(fp);
    }

//          ORDERSTATUS

    if ((Term.pClientData[iTermId].OrderStatusDataPtr =
(ORDER_STATUS_DATA *)tpalloc("CARRAY", NULL, sizeof(ORDER_STATUS_DATA))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> Init Thread %d :

OrderStatus tpalloc failed: iRc = %d \r\n",

                GetCurrentThreadId(), iRc);

```

```

        fclose(fp);
    }
    sprintf(buf, "Tpccalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO,
buf, 0, 0);
    return TRUE;
}
if ( dLog )
{
    FILE *fp;
    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** Init Thread %d iTermId %d *
OrderStatusDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].OrderStatusDataPtr);
    fclose(fp);
}
// DELIVERY
if ((Term.pClientData[iTermId].DeliveryDataPtr = (DELIVERY_DATA
*)tpalloc("CARRY", NULL, sizeof(DELIVERY_DATA))) == NULL)
{
    iRc = tperrno;
    {
        FILE *fp;
        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, ">>>> Init Thread %d : Delivery
tpalloc failed: iRc = %d \r\n",
            GetCurrentThreadId(), iRc);
        fclose(fp);
    }
    sprintf(buf, "Tpccalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO,
buf, 0, 0);
    return TRUE;
}
if ( dLog )
{
    FILE *fp;
    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** Init Thread %d iTermId %d *
DeliveryDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].DeliveryDataPtr);
    fclose(fp);
}
// STOCKLEVEL
if ((Term.pClientData[iTermId].StockLevelDataPtr =
(STOCK_LEVEL_DATA *)tpalloc("CARRY", NULL, sizeof(STOCK_LEVEL_DATA))) == NULL)
{
    iRc = tperrno;
    {
        FILE *fp;

```

```

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, ">>>> Init Thread %d :
StockLevel tpalloc failed: iRc = %d \r\n",
            GetCurrentThreadId(), iRc);
        fclose(fp);
    }
    sprintf(buf, "Tpccalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO,
buf, 0, 0);
    return TRUE;
}
if ( dLog )
{
    FILE *fp;
    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** Init Thread %d iTermId %d *
StockLevelDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].StockLevelDataPtr);
    fclose(fp);
}
#endif
return FALSE;
}
/* FUNCTION: BOOL Close(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId)
 *
 * PURPOSE: This function closes the sql connection for use.
 *
 * ARGUMENTS: EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
 *
 * int iTermId id of browser client that this connection is for.
 * int iSyncId sync id of client browser
 *
 * RETURNS: BOOL FALSE if successful
 * TRUE if an error
occurs and connection cannot be terminated.
 *
 * COMMENTS: None
 */
static BOOL Close(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int iSyncId)
{
    PECBINFO pEcbInfo;
    if (Term.pClientData[iTermId].dbproc != NULL)
    {
        if ( (pEcbInfo =
(PECBINFO)dbgetuserdata(Term.pClientData[iTermId].dbproc)) )
        {
            pEcbInfo->iTermId = -1;
            pEcbInfo->iSyncId = -1;
            free(pEcbInfo); //free up user info
        }
    }
    // return SQLCloseConnection(pECB,
Term.pClientData[iTermId].dbproc);

```

```

        return 1;
    }
    UNUSEDPARAM(iSyncId);
}

/* FUNCTION: void FormatString(char *szDest, char *szPic, char *szSrc)
 *
 * PURPOSE:      This function formats a character string for inclusion in the
 *               HTML formatted page being constructed.
 *
 * ARGUMENTS:   char          *szDest  Destination buffer where formatted
 *               string is to be placed
 *               char          *szPic   picture string
 *               which describes how character value is to be
 *               formatted.
 *               char          *szSrc   character
 *               string value.
 *
 * RETURNS:     None
 *
 * COMMENTS:    This functions is used to format TPC-C phone and zip value
 *               strings.
 */
static void FormatString(char *szDest, char *szPic, char *szSrc)
{
    while( *szPic )
    {
        if ( *szPic == 'X' )
        {
            if ( *szSrc )
                *szDest++ = *szSrc++;
            else
                *szDest++ = ' ';
        }
        else
            *szDest++ = *szPic;
        szPic++;
    }
    *szDest = 0;

    return;
}

/* FUNCTION: char *MakeStockLevelForm(int iTermId, int iSyncId, BOOL bInput)
 *
 * PURPOSE:      This function constructs the Stock Level HTML page.
 *
 * ARGUMENTS:   int          iTermId  client browser
 *               terminal id
 *               int          iSyncId  client browser sync id
 *               BOOL        bInput   TRUE
 *
 * if form is being constructed for input else FALSE
 *
 * RETURNS:     char *          A pointer to buffer
 *               inside client structure where HTML form is built.
 */

```

```

 * COMMENTS:      The internal client buffer is created when the terminal id is
 *               assigned and should not
 *               be freed except when the client terminal id
 *               is no longer needed.
 */
static char *MakeStockLevelForm(int iTermId, int iSyncId, BOOL bInput)
{
    char          *szForm;

    szForm = (char *)Term.pClientData[iTermId].szBuffer;

    Term.pClientData[iTermId].StockLevelData.w_id      =
(short)Term.pClientData[iTermId].w_id;
    Term.pClientData[iTermId].StockLevelData.d_id      =
(short)Term.pClientData[iTermId].d_id;
    Term.pClientData[iTermId].StockLevelData.num_deadlocks = 0;

    if ( bInput)
    {
        strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"7\">"; // STOCK_LEVEL_FORM = 7

        wprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iTermId, iSyncId);

        strcat(szForm, "<PRE>
Level<BR>");

        wprintf(szForm+strlen(szForm), "Warehouse: %4.4d District:
%2.2d<BR><BR>", Term.pClientData[iTermId].StockLevelData.w_id,
Term.pClientData[iTermId].StockLevelData.d_id);

        strcat(szForm, "Stock Level Threshold: <INPUT NAME=\"TT*\"
SIZE=2><BR><BR>\
low stock: <BR><HR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">\
</FORM></HTML>");

    }
    else // Not bInput
    {
        strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"7\">"; // STOCK_LEVEL_FORM = 7

        wprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">", iTermId, iSyncId);

        strcat(szForm, "<PRE>
Level<BR>");
    }
}

```

```

        wsprintf(szForm+strlen(szForm), "Warehouse: %4.4d District:
%2.2d<BR><BR>\
Stock Level Threshold: %2.2d<BR><BR>low stock: %3.3d</PRE><BR><HR>",
        Term.pClientData[iTermId].StockLevelData.w_id,
        Term.pClientData[iTermId].StockLevelData.d_id,
        Term.pClientData[iTermId].StockLevelData.thresh_hold,
        Term.pClientData[iTermId].StockLevelData.low_stock);
        strcat(szForm, "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..NewOrder..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">\
</FORM></HTML>");
    }
    return szForm;
}
/* FUNCTION: char *MakeMainMenuForm(int iTermId, int iSyncId)
 * PURPOSE: This function
 * ARGUMENTS: int iTermId client browser
terminal id
 * int iSyncId client browser sync id
 * RETURNS: char * A pointer to buffer
inside client structure where HTML form is built.
 * COMMENTS: The internal client buffer is created when the terminal id is
assigned and should not
 * be freed except when the client terminal id
is no longer needed.
 */
static char *MakeMainMenuForm(int iTermId, int iSyncId)
{
    char *szForm;
    szForm = (char *)Term.pClientData[iTermId].szBuffer;
    strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C Main
Menu</TITLE></HEAD><BODY>"
    "Select Desired
Transaction.<BR><HR>"
    "<FORM ACTION=\"tpcc.dll\"
METHOD=\"GET\">");
    strcat(szForm, "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">");
    strcat(szForm, "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">");
    wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\" NAME=\"TERMINID\"
VALUE=\"%d\">", iTermId);
    wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\"
VALUE=\"%d\">", iSyncId);

```

```

        wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\" NAME=\"FORMID\"
VALUE=\"%d\">", MAIN_MENU_FORM);
        strcat(szForm, "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..NewOrder..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Payment..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Delivery..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Order-Status..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Exit..\">"
        "</FORM>"
        "</HTML>");
    }
    return szForm;
}
/* FUNCTION: char *MakeWelcomeForm(void)
 * PURPOSE: This function
 * ARGUMENTS: None
 * RETURNS: char * A pointer to the static
HTML welcome form.
 * COMMENTS: The welcome form is static.
 */
static char *MakeWelcomeForm(void)
{
    return szWelcomeForm;
}
/* FUNCTION: char *MakeNewOrderForm(int iTermId, BOOL bInput, BOOL bValid)
 * PURPOSE: This function
 * ARGUMENTS: int iTermId client browser
terminal id
 * int iSyncId client browser sync id
 * BOOL bInput TRUE
if form is being constructed for input else FALSE
 * BOOL bValid TRUE
if NewOrderData valid, ELSE FALSE effects output only
 * RETURNS: char * A pointer to buffer
inside client structure where HTML form is built.
 * COMMENTS: The internal client buffer is created when the terminal id is
assigned and should not
 * be freed except when the client terminal id
is no longer needed.
 */
static char *MakeNewOrderForm(int iTermId, int iSyncId, BOOL bInput, BOOL bValid)
{
    char *szForm;
    // char szName[146];

```



```

//      char      szCredit[14];
//      int       i;

      szForm = (char *)Term.pClientData[iTermId].szBuffer;

      Term.pClientData[iTermId].NewOrderData.w_id =
Term.pClientData[iTermId].w_id;

      if ( bInput)
      {
          strcpy(szForm,"<HTML><HEAD><TITLE>TPC-C New Order</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"3\">"); // NEW_ORDER_FORM = 3

          sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">\
<PRE>
New Order<BR>\
Warehouse: %4.4d District: <INPUT NAME=\"DID*\" SIZE=1>
Date:<BR>\
Customer: <INPUT NAME=\"CID*\" SIZE=4> Name:
Credit:
%Disc:<BR>\
Order Number:          Number of Lines:          W_tax:          D_tax:<BR><BR>\
  Supp_W Item_Id Item Name          Qty Stock B/G Price Amount<BR>\",
iTermId,
iSyncId,

Term.pClientData[iTermId].NewOrderData.w_id);

          strcat(szForm,
NAME=\"IID00*\" SIZE=6>
SIZE=1><BR>\
<INPUT NAME=\"SP01*\" SIZE=4> <INPUT NAME=\"IID01*\" SIZE=6>
<INPUT NAME=\"Qty01*\" SIZE=1><BR>\
<INPUT NAME=\"SP02*\" SIZE=4> <INPUT NAME=\"IID02*\" SIZE=6>
<INPUT NAME=\"Qty02*\" SIZE=1><BR>\
<INPUT NAME=\"SP03*\" SIZE=4> <INPUT NAME=\"IID03*\" SIZE=6>
<INPUT NAME=\"Qty03*\" SIZE=1><BR>\
<INPUT NAME=\"SP04*\" SIZE=4> <INPUT NAME=\"IID04*\" SIZE=6>
<INPUT NAME=\"Qty04*\" SIZE=1><BR>\
<INPUT NAME=\"SP05*\" SIZE=4> <INPUT NAME=\"IID05*\" SIZE=6>
<INPUT NAME=\"Qty05*\" SIZE=1><BR>\
<INPUT NAME=\"SP06*\" SIZE=4> <INPUT NAME=\"IID06*\" SIZE=6>
<INPUT NAME=\"Qty06*\" SIZE=1><BR>\
<INPUT NAME=\"SP07*\" SIZE=4> <INPUT NAME=\"IID07*\" SIZE=6>
<INPUT NAME=\"Qty07*\" SIZE=1><BR>\
<INPUT NAME=\"SP08*\" SIZE=4> <INPUT NAME=\"IID08*\" SIZE=6>
<INPUT NAME=\"Qty08*\" SIZE=1><BR>\
<INPUT NAME=\"SP09*\" SIZE=4> <INPUT NAME=\"IID09*\" SIZE=6>
<INPUT NAME=\"Qty09*\" SIZE=1><BR>\
<INPUT NAME=\"SP10*\" SIZE=4> <INPUT NAME=\"IID10*\" SIZE=6>
<INPUT NAME=\"Qty10*\" SIZE=1><BR>\
<INPUT NAME=\"SP11*\" SIZE=4> <INPUT NAME=\"IID11*\" SIZE=6>
<INPUT NAME=\"Qty11*\" SIZE=1><BR>\
<INPUT NAME=\"SP12*\" SIZE=4> <INPUT NAME=\"IID12*\" SIZE=6>
<INPUT NAME=\"Qty12*\" SIZE=1><BR>\
<INPUT NAME=\"SP13*\" SIZE=4> <INPUT NAME=\"IID13*\" SIZE=6>
<INPUT NAME=\"Qty13*\" SIZE=1><BR>\
<INPUT NAME=\"SP14*\" SIZE=4> <INPUT NAME=\"IID14*\" SIZE=6>
<INPUT NAME=\"Qty14*\" SIZE=1><BR>\

```

```

Execution Status:
Total:<BR><HR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">\
</FORM></HTML>" );

      }
      else // not bInput
      {
          if (bValid)
          {
              //      FormatHTMLString(szName,
Term.pClientData[iTermId].NewOrderData.c_last, 16),
              //      FormatHTMLString(szCredit,
Term.pClientData[iTermId].NewOrderData.c_credit, 2);

              strcpy(szForm,"<HTML><HEAD><TITLE>TPC-C New Order</TITLE></HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"3\">"); // NEW_ORDER_FORM = 3

              sprintf(szForm+strlen(szForm),"<INPUT TYPE=\"hidden\" NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">\
<PRE>
New Order<BR>\
Warehouse: %4.4d District: %2.2d          Date: %2.2d-%2.2d-%4.4d
%2.2d:%2.2d:%2.2d <BR>\
Customer: %4.4d Name: %s Credit: %s %Disc: %5.2f          <BR>\
Order Number: %8.8d Number of Lines: %2.2d          W_tax: %5.2f D_tax: %5.2f
<BR><BR>\
  Supp_W Item_Id Item Name          Qty Stock B/G Price Amount<BR>\",
iTermId,
iSyncId,
Term.pClientData[iTermId].NewOrderData.w_id,
Term.pClientData[iTermId].NewOrderData.d_id,

Term.pClientData[iTermId].NewOrderData.o_entry_d.day,
Term.pClientData[iTermId].NewOrderData.o_entry_d.month,
Term.pClientData[iTermId].NewOrderData.o_entry_d.year,
Term.pClientData[iTermId].NewOrderData.o_entry_d.hour,
Term.pClientData[iTermId].NewOrderData.o_entry_d.minute,
Term.pClientData[iTermId].NewOrderData.o_entry_d.second,
Term.pClientData[iTermId].NewOrderData.c_id,
Term.pClientData[iTermId].NewOrderData.c_last,
Term.pClientData[iTermId].NewOrderData.c_credit,
Term.pClientData[iTermId].NewOrderData.c_discount*100.0,
Term.pClientData[iTermId].NewOrderData.o_id,
Term.pClientData[iTermId].NewOrderData.o_ol_cnt,
Term.pClientData[iTermId].NewOrderData.w_tax*100.0,
Term.pClientData[iTermId].NewOrderData.d_tax*100.0);

```

```

// We could make 0-4 be hard coded in one sprintf, since we know that there is at
// least 5
// order lines, or is this a benchmark special??
// etw
        for(i=0; i<Term.pClientData[iTermId].NewOrderData.o_ol_cnt; i++)
        {
            FormatHTMLString(szName,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_i_name, 24);

            sprintf(szForm+strlen(szForm), " %4.4d
%6.6d %24s %2.2d %3.3d %1.1s %6.2f %7.2f <BR>",
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_supply_w_id,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_i_id,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_i_name,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_quantity,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_stock,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_brand_generic,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_i_price,
Term.pClientData[iTermId].NewOrderData.Ol[i].ol_amount );
        }
        for(; i<15; i++)
            strcat(szForm, "<BR>");
        sprintf(szForm+strlen(szForm), "Execution Status:
%24.24s Total: %8.2f ",
Term.pClientData[iTermId].NewOrderData.execution_status,
Term.pClientData[iTermId].NewOrderData.total_amount);

        strcat(szForm, "</PRE><HR><BR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">\
</FORM></HTML>");
    }
    else // not bValid
    {
        // FormatHTMLString(szName,
Term.pClientData[iTermId].NewOrderData.c_last, 16),
        // FormatHTMLString(szCredit,
Term.pClientData[iTermId].NewOrderData.c_credit, 2);

        strcpy(szForm, "<HTML><HEAD><TITLE>TPC-C New Order</TITLE>>\
</HEAD><BODY><FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"1\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"1\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"3\">"; // NEW_ORDER_FORM = 3
    }
}

```

```

        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">\
<PRE>
New Order<BR>\
Warehouse: %4.4d District: %2.2d Date:<BR>\
Customer: %4.4d Name: %s Credit: %s %Disc:<BR>\
Order Number: %8.8d Number of Lines: W_tax: D_tax:<BR><BR>",
iTermId,
iSyncId,
Term.pClientData[iTermId].NewOrderData.w_id,
Term.pClientData[iTermId].NewOrderData.d_id,
Term.pClientData[iTermId].NewOrderData.c_id,

Term.pClientData[iTermId].NewOrderData.c_last,

Term.pClientData[iTermId].NewOrderData.c_credit,
Term.pClientData[iTermId].NewOrderData.o_id
);

        strcat(szForm, " Supp_W Item_Id Item Name Qty
Stock B/G Price Amount<BR>\
<BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR><BR>");

        sprintf(szForm+strlen(szForm), "Execution Status: %24.24s
Total:",
Term.pClientData[iTermId].NewOrderData.execution_status);

        strcat(szForm, "</PRE><HR><BR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">\
</FORM></HTML>");
    }
}
return szForm;
}

/* FUNCTION: char *MakePaymentForm(int iTermId, int iSyncId, BOOL bInput)
*
* PURPOSE: This function
*
* ARGUMENTS: int iTermId client browser
terminal id
* int
iSyncId client browser sync id
* BOOL bInput TRUE
if form is being constructed for input else FALSE
*
* RETURNS: char * A pointer to buffer
inside client structure where HTML form is built.
*
* COMMENTS: The internal client buffer is created when the terminal id is
assigned and should not
* be freed except when the client terminal id
is no longer needed.
*/

static char *MakePaymentForm(int iTermId, int iSyncId, BOOL bInput)

```

```

{
    char      *szForm;
    char      *ptr;
    char      szTmp[64];
    char      szW_Zip[26];
    char      szD_Zip[26];
    char      szC_Zip[26];
    char      szC_Phone[26];
    // char    szTmpStr1[122];
    // char    szTmpStr2[122];
    // char    szTmpStr3[122];
    // char    szTmpStr4[122];
    int       i;
    int       l;
    char      *szZipPic = "XXXX-XXXX";

    szForm = (char *)Term.pClientData[iTermId].szBuffer;

    Term.pClientData[iTermId].PaymentData.w_id =
Term.pClientData[iTermId].w_id;

    if ( bInput )
    {
        strcpy(szForm,          "<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"4\">"; // PAYMENT_FORM = 4

        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMIN\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNID\" VALUE=\"%d\">\
<PRE>                                Payment<BR>\
Date:<BR><BR>\
Warehouse: %4.4d",

                                iTermId,
                                iSyncId,

                                Term.pClientData[iTermId].PaymentData.w_id);

        strcat(szForm,
"
        District: <INPUT NAME=\"DID\"\"
SIZE=1><BR><BR><BR><BR><BR>\
Customer: <INPUT NAME=\"CID\"\" SIZE=4>\
Cust-Warehouse: <INPUT NAME=\"CWI\"\" SIZE=4> \
Cust-District: <INPUT NAME=\"CDI\"\" SIZE=1><BR>\
Name:                                <INPUT NAME=\"CLT\"\" SIZE=16>
Since:<BR>\

                                Credit:<BR>\
                                Disc:<BR>\
                                Phone:<BR><BR>\
Amount Paid:          $<INPUT NAME=\"HAM\"\" SIZE=7>          New Cust Balance:<BR>\
Credit Limit:<BR><BR>Cust-Data: <BR><BR><BR><BR></PRE><HR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\"><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">\
</BODY></FORM></HTML>");
    }
    else // Not bInput
    {

```

```

//          FormatHTMLString(szTmpStr1,
Term.pClientData[iTermId].PaymentData.w_street_1, 20);
//          FormatHTMLString(szTmpStr2,
Term.pClientData[iTermId].PaymentData.d_street_1, 20);

//          FormatHTMLString(szTmpStr3,
Term.pClientData[iTermId].PaymentData.w_street_2, 20);
//          FormatHTMLString(szTmpStr4,
Term.pClientData[iTermId].PaymentData.d_street_2, 20);

        FormatString(szW_Zip, szZipPic,
Term.pClientData[iTermId].PaymentData.w_zip);
        FormatString(szD_Zip, szZipPic,
Term.pClientData[iTermId].PaymentData.d_zip);

//          FormatHTMLString(szTmpStr5,
Term.pClientData[iTermId].PaymentData.w_city, 20);
//          FormatHTMLString(szTmpStr6,
Term.pClientData[iTermId].PaymentData.w_state, 2);
//          FormatHTMLString(szTmpStr7,
Term.pClientData[iTermId].PaymentData.d_city, 20);
//          FormatHTMLString(szTmpStr8,
Term.pClientData[iTermId].PaymentData.d_state, 2);

//          FormatHTMLString(szTmpStr9,
Term.pClientData[iTermId].PaymentData.c_first, 16);
//          FormatHTMLString(szTmpStr10,
Term.pClientData[iTermId].PaymentData.c_middle, 2);
//          FormatHTMLString(szTmpStr11,
Term.pClientData[iTermId].PaymentData.c_last, 16);

//          FormatHTMLString(szTmpStr12,
Term.pClientData[iTermId].PaymentData.c_street_1, 20);
//          FormatHTMLString(szTmpStr13,
Term.pClientData[iTermId].PaymentData.c_credit, 2);

//          FormatHTMLString(szTmpStr14,
Term.pClientData[iTermId].PaymentData.d_street_2, 20);

        FormatString(szC_Zip, szZipPic,
Term.pClientData[iTermId].PaymentData.c_zip);
        FormatString(szC_Phone, "XXXXXX-XXX-XXX-XXXX",
Term.pClientData[iTermId].PaymentData.c_phone);

//          FormatHTMLString(szTmpStr15,
Term.pClientData[iTermId].PaymentData.c_city, 20);
//          FormatHTMLString(szTmpStr16,
Term.pClientData[iTermId].PaymentData.c_state, 2);

        strcpy(szForm,          "<HTML><HEAD><TITLE>TPC-C
Payment</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"4\">"; // PAYMENT_FORM = 4

        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMIN\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNID\" VALUE=\"%d\">\
<PRE>                                Payment<BR>\
Date: %2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d <BR><BR>\
Warehouse: %4.4d          District: %2.2d<BR>\
%20s          %20s<BR>\

```

```

%-20s          %-20s<BR>\
%-20s %-2s %10.10s %-20s %-2s %10.10s<BR><BR>\
Customer: %4.4d Cust-Warehouse: %4.4d Cust-District: %2.2d<BR>\
Name: %-16s %-2s %-16s Since: %2.2d-%2.2d-%4.4d<BR>\
      %-20s          Credit: %s<BR>\
      %-20s          %%Disc: %5.2f<BR>\
      %-20s %-2s %10.10s Phone: %-19.19s<BR><BR>\
Amount Paid:      $%7.2f      New Cust Balance: $%14.2f<BR>\
Credit Limit:    $%13.2f<BR><BR>,

    iTermId,
    iSyncId,
    Term.pClientData[iTermId].PaymentData.h_date.day,
    Term.pClientData[iTermId].PaymentData.h_date.month,
    Term.pClientData[iTermId].PaymentData.h_date.year,
    Term.pClientData[iTermId].PaymentData.h_date.hour,
    Term.pClientData[iTermId].PaymentData.h_date.minute,
    Term.pClientData[iTermId].PaymentData.h_date.second,
    Term.pClientData[iTermId].PaymentData.w_id,
    Term.pClientData[iTermId].PaymentData.d_id,
    Term.pClientData[iTermId].PaymentData.w_street_1,
    Term.pClientData[iTermId].PaymentData.d_street_1,
    Term.pClientData[iTermId].PaymentData.w_street_2,
    Term.pClientData[iTermId].PaymentData.d_street_2,
    Term.pClientData[iTermId].PaymentData.w_city,
    Term.pClientData[iTermId].PaymentData.w_state,
    szW_Zip,
    Term.pClientData[iTermId].PaymentData.d_city,
    Term.pClientData[iTermId].PaymentData.d_state,
    szD_Zip,
    Term.pClientData[iTermId].PaymentData.c_id,
    Term.pClientData[iTermId].PaymentData.c_w_id,
    Term.pClientData[iTermId].PaymentData.c_d_id,
    Term.pClientData[iTermId].PaymentData.c_first,
    Term.pClientData[iTermId].PaymentData.c_middle,
    Term.pClientData[iTermId].PaymentData.c_last,
    Term.pClientData[iTermId].PaymentData.c_since.day,
    Term.pClientData[iTermId].PaymentData.c_since.month,
    Term.pClientData[iTermId].PaymentData.c_since.year,
    Term.pClientData[iTermId].PaymentData.c_street_1,
    Term.pClientData[iTermId].PaymentData.c_credit,
    Term.pClientData[iTermId].PaymentData.d_street_2,

    Term.pClientData[iTermId].PaymentData.c_discount*100.0,
    Term.pClientData[iTermId].PaymentData.c_city,
    Term.pClientData[iTermId].PaymentData.c_state,
    szC_Zip, szC_Phone,
    Term.pClientData[iTermId].PaymentData.h_amount,
    Term.pClientData[iTermId].PaymentData.c_balance,
    Term.pClientData[iTermId].PaymentData.c_credit_lim);

// This part can probably be made a little more efficient
//etw
    ptr = Term.pClientData[iTermId].PaymentData.c_credit;
    if ( *ptr == 'B' && *(ptr+1) == 'C' )
    {
        ptr = Term.pClientData[iTermId].PaymentData.c_data;
        l = strlen( ptr ) / 50;
        for(i=0; i<4; i++, ptr += 50)
        {
            if ( i <= 1 )
                UtilStrCpy(szTmp, ptr, 50);
            else

```

```

        szTmp[0] = 0;
        if ( !i )
        {
            FormatHTMLString(szTmpStr1, szTmp,
                wsprintf(szForm+strlen(szForm),
                    "Cust-Data: %-50s<BR>", szTmp);
        }
        else
        {
            FormatHTMLString(szTmpStr1, szTmp,
                wsprintf(szForm+strlen(szForm), "
        }
    }
}
else
    strcat(szForm, "Cust-Data: <BR><BR><BR><BR>");
    strcat(szForm, "</PRE><HR><BR>\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">\
</BODY></FORM></HTML>");
}
return szForm;
}
/* FUNCTION: char *MakeOrderStatusForm(int iTermId, int iSyncId, BOOL bInput)
 * PURPOSE:      This function
 * ARGUMENTS:    int iTermId client browser
                 terminal id
                 int iSyncId client browser sync id
                 BOOL bInput TRUE
if form is being constructed for input else FALSE
 * RETURNS:      char * A pointer to buffer
                 inside client structure where HTML form is built.
 * COMMENTS:     The internal client buffer is created when the terminal id is
                 assigned and should not be freed except when the client terminal id
                 is no longer needed.
 */
static char *MakeOrderStatusForm(int iTermId, int iSyncId, BOOL bInput)
{
    char *szForm;
    // char c_first[98];
    // char c_middle[14];
    // char c_last[98];
    int i;
    szForm = (char *)Term.pClientData[iTermId].szBuffer;

```

```

Term.pClientData[iTermId].OrderStatusData.w_id =
Term.pClientData[iTermId].w_id;

if ( bInput )
{
    strcpy(szForm,          "<HTML><HEAD><TITLE>TPC-C Order-
Status</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"6\">"); // ORDER_STATUS_FORM = 6

    wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
            iTermId,
            iSyncId);

    strcat(szForm,          "<PRE>                                Order-
Status<BR>");
    wsprintf(szForm+strlen(szForm), "Warehouse: %4.4d ",
Term.pClientData[iTermId].OrderStatusData.w_id);

    strcat(szForm,          "District: <INPUT NAME=\"DID*\" SIZE=1><BR>\
Customer: <INPUT NAME=\"CID*\" SIZE=4>   Name:                <INPUT
NAME=\"CLT*\" SIZE=23><BR>\
Cust-Balance:<BR><BR>\
Order-Number:                Entry-Date:                Carrier-Number:<BR>\
Supply-W   Item-Id   Qty   Amount   Delivery-Date<BR></PRE>\
<HR><INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\"><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">\
</BODY></FORM></HTML>");
}
else // Not bInput
{
    strcpy(szForm,          "<HTML><HEAD><TITLE>TPC-C Order-
Status</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"6\">"); // ORDER_STATUS_FORM = 6

    wsprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
            iTermId,
            iSyncId);

    strcat(szForm,          "<PRE>                                Order-Status<BR>"
);

    sprintf(szForm+strlen(szForm), "Warehouse: %4.4d   District: %2.2d<BR>\
Customer: %4.4d   Name: %-16s %-2s %-16s<BR>\
Cust-Balance: %9.2f<BR><BR>\
Order-Number: %8.8d   Entry-Date: %2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d   Carrier-
Number: %2.2d<BR>",
Term.pClientData[iTermId].OrderStatusData.w_id,
Term.pClientData[iTermId].OrderStatusData.d_id,

```

```

Term.pClientData[iTermId].OrderStatusData.c_id,
Term.pClientData[iTermId].OrderStatusData.c_first,
Term.pClientData[iTermId].OrderStatusData.c_middle,
Term.pClientData[iTermId].OrderStatusData.c_last,
Term.pClientData[iTermId].OrderStatusData.c_balance,
Term.pClientData[iTermId].OrderStatusData.o_id,
Term.pClientData[iTermId].OrderStatusData.o_entry_d.day,
Term.pClientData[iTermId].OrderStatusData.o_entry_d.month,
Term.pClientData[iTermId].OrderStatusData.o_entry_d.year,
Term.pClientData[iTermId].OrderStatusData.o_entry_d.hour,
Term.pClientData[iTermId].OrderStatusData.o_entry_d.minute,
Term.pClientData[iTermId].OrderStatusData.o_entry_d.second,
Term.pClientData[iTermId].OrderStatusData.o_carrier_id);

    strcat(szForm+strlen(szForm), "Supply-W   Item-Id   Qty
Amount   Delivery-Date<BR>");
    for(i=0; i<Term.pClientData[iTermId].OrderStatusData.o_ol_cnt;
i++)
    {
        sprintf(szForm+strlen(szForm), "   %4.4d   %6.6d
%2.2d   %$8.2f   %2.2d-%2.2d-%4.4d<BR>",
Term.pClientData[iTermId].OrderStatusData.OlOrderStatusData[i].ol_supply_w
_id,
Term.pClientData[iTermId].OrderStatusData.OlOrderStatusData[i].ol_i_id,
Term.pClientData[iTermId].OrderStatusData.OlOrderStatusData[i].ol_quantity
,
Term.pClientData[iTermId].OrderStatusData.OlOrderStatusData[i].ol_amount,
Term.pClientData[iTermId].OrderStatusData.OlOrderStatusData[i].ol_delivery
_d.day,
Term.pClientData[iTermId].OrderStatusData.OlOrderStatusData[i].ol_delivery
_d.month,
Term.pClientData[iTermId].OrderStatusData.OlOrderStatusData[i].ol_delivery
_d.year);
    }

    strcat(szForm,          "<BR></PRE><HR><INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"\"..NewOrder..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"\"..Payment..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"\"..Delivery..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"\"..Order-Status..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"\"..Stock-Level..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"\"..Exit..\">\

```

```

</BODY></FORM></HTML>" );
    }

    return szForm;
}

/* FUNCTION: char *MakeDeliveryForm(int iTermId, int iSyncId, BOOL bInput)
 *
 * PURPOSE:      This function
 *
 * ARGUMENTS:   int          iTermId   client browser
                terminal id
 *
 *              int          iSyncId   client browser sync id
 *              BOOL        bInput    TRUE
if form is being constructed for input else FALSE
 *
 * RETURNS:     char *        A pointer to buffer
                inside client structure where HTML form is built.
 *
 * COMMENTS:    The internal client buffer is created when the terminal id is
                assigned and should not
                be freed except when the client terminal id
                is no longer needed.
 */

static char *MakeDeliveryForm(int iTermId, int iSyncId, BOOL bInput)
{
    char      *szForm;

    szForm = (char *)Term.pClientData[iTermId].szBuffer;

    Term.pClientData[iTermId].DeliveryData.w_id =
Term.pClientData[iTermId].w_id;

    if ( bInput )
    {
        strcpy( szForm,          "<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"PI*\" VALUE=\"\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"5\">"); // DELIVERY_FORM = 5

        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",

                iTermId,

                iSyncId);

        strcat(szForm,          "<PRE>
Delivery<BR>" );

        sprintf(szForm+strlen(szForm), "Warehouse: %4.4d<BR><BR>",
Term.pClientData[iTermId].DeliveryData.w_id);

        strcat( szForm, "Carrier Number: <INPUT NAME=\"OCD*\"
SIZE=1><BR><BR>\
Execution Status:<BR></PRE>\
<HR><INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Process\">\

```

```

<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Menu\">\
</BODY></FORM></HTML>" );

    }
    else // Not bInput
    {
        strcpy( szForm,          "<HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD><BODY>\
<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">\
<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">\
<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"5\">"); // DELIVERY_FORM = 5

        sprintf(szForm+strlen(szForm), "<INPUT TYPE=\"hidden\"
NAME=\"TERMINID\" VALUE=\"%d\">\
<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",

                iTermId,

                iSyncId);

        strcat(szForm,          "<PRE>
Delivery<BR>" );

        sprintf(szForm+strlen(szForm), "Warehouse: %4.4d<BR><BR>",
Term.pClientData[iTermId].DeliveryData.w_id);

        sprintf(szForm+strlen(szForm), "Carrier Number: %2.2d<BR><BR>\
Execution Status: %25.25s<BR></PRE>",

                Term.pClientData[iTermId].DeliveryData.o_carrier_id,

                Term.pClientData[iTermId].DeliveryData.execution_status);

        strcat(szForm,          "<HR><INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"..NewOrder..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Payment..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Delivery..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-Status..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-Level..\">\
<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">\
</BODY></FORM></HTML>" );

    }

    return szForm;
}

/* FUNCTION: void UtilStrCpy(char * pDest, char * pSrc, int n)
 *
 * PURPOSE:      This function copies n characters from string pSrc to pDst and
                places a
                null character at the end of the destination string.
 *
 * ARGUMENTS:   char          *pDest   destination string
                pointer
 *              char          *pSrc
 *              int          n
                number of characters to copy
 *
 * RETURNS:     None

```

```

*
* COMMENTS:      Unlike strncpy this function ensures that the result string is
*                always null terminated.
*
*/

static void UtilStrCpy(char * pDest, char * pSrc, int n)
{
    strncpy(pDest, pSrc, n);
    pDest[n] = '\0';

    return;
}

/* FUNCTION: void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int
iTermId, int iSyncId)
*
* PURPOSE:      This function gets and validates the input data from the new
order form
*                filling in the required input variables. it then calls
the SQLNewOrder
*                transaction, constructs the output form and writes it
back to client
*                browser.
*
* ARGUMENTS:    EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*                int iTermId client browser terminal id
*                int iSyncId client browser sync id
*
* RETURNS:      None
*
* COMMENTS:      None
*/

static void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId)
{
    int iRc, iError;
    long ilen, *olen;
    // PECBINFO pEcbInfo;
    char buf[40];
#ifdef LOCAL_ALLOC
    NEW_ORDER_DATA *NewOrderDataPtr;
#endif

    if((iRc = ThrTpInit(pECB)) < 0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessNewOrder Thread %d iTermId %d
Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);
        }
        sprintf(buf, "NewOrder ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());

```

```

        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
    }

    memset(&Term.pClientData[iTermId].NewOrderData, 0,
sizeof(NEW_ORDER_DATA));

    Term.pClientData[iTermId].NewOrderData.w_id =
Term.pClientData[iTermId].w_id;

    if ( (iError=GetNewOrderData(pECB->lpszQueryString,
&Term.pClientData[iTermId].NewOrderData)) != ERR_SUCCESS )
    {
        ErrorMessage(pECB, iError, ERR_TYPE_WEBDLL, NULL, iTermId,
iSyncId);
        return;
    }

#ifdef LOCAL_ALLOC
    if ((NewOrderDataPtr = (NEW_ORDER_DATA *) tmalloc("CARRAY", NULL,
sizeof(NEW_ORDER_DATA))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessNewOrder Thread %d iTermId %d
tpalloc tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "NewOrder Tmalloc %d", iRc);
        ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
        tfree((char *)NewOrderDataPtr);
        return;
    }

    *NewOrderDataPtr = Term.pClientData[iTermId].NewOrderData;
    ilen = sizeof(NEW_ORDER_DATA);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessNewOrder Thread %d iTermId %d
NewOrderDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId, &NewOrderDataPtr);
        fclose(fp);
    }

    if (tpcall("NEWORDER", (char *)NewOrderDataPtr, ilen, (char
**)&NewOrderDataPtr, (long *)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessNewOrder Thread %d iTermId %d
tpcall tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);

```

```

        fclose(fp);
    }
    sprintf(buf, "Neworder tpcall %d", iRc);
    ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
    tpfree((char *)NewOrderDataPtr);
return;
}

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "*** ProcessNewOrder Thread %d iTermId %d
NewOrderDataPtr: %x \r\n",
        GetCurrentThreadId(), iTermId, &NewOrderDataPtr);
    fclose(fp);
}

Term.pClientData[iTermId].NewOrderData = *NewOrderDataPtr;

iRc = NewOrderDataPtr->retval;

    tpfree((char *)NewOrderDataPtr);

#else

    *Term.pClientData[iTermId].NewOrderDataPtr =
Term.pClientData[iTermId].NewOrderData;

    ilen = sizeof(NEW_ORDER_DATA);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessNewOrder Thread %d iTermId %d
NewOrderDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].NewOrderDataPtr);
        fclose(fp);
    }

    if (tpcall("NEWORDER", (char *)Term.pClientData[iTermId].NewOrderDataPtr, ilen,
(char **)&Term.pClientData[iTermId].NewOrderDataPtr, (long
*)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "*** ProcessNewOrder Thread %d iTermId %d
tpcall tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Neworder tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
        tpfree((char *)NewOrderDataPtr);
//
return;

```

```

    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessNewOrder Thread %d iTermId %d
NewOrderDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].NewOrderDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].NewOrderData =
*Term.pClientData[iTermId].NewOrderDataPtr;

    iRc = Term.pClientData[iTermId].NewOrderDataPtr->retval;

#endif
#ifdef BOIL
    Sleep( 1100 );
#endif
    if ( iRc < 0 )
        ErrorMessage(pECB, ERR_NEW_ORDER_NOT_PROCESSED, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
    else
        WriteZString(pECB, MakeNewOrderForm(iTermId, iSyncId, FALSE,
(BOOL)iRc) );
    return;
}

/* FUNCTION: void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId)
*
* PURPOSE:          This function gets and validates the input data from the payment
form
*                   filling in the required input variables. It then calls
the SQLPayment
*                   transaction, constructs the output form and writes it
back to client
*                   browser.
*
* ARGUMENTS:       EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetsrv.
*                   int
iTermId client browser terminal id
*                   int
iSyncId client browser sync id
*
* RETURNS:         None
*
* COMMENTS:       None
*/

static void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId)
{
    int iRc, iError;
    long ilen, *olen;
    PECBINFO pEcbInfo;
    char buf[40];

```



```

#ifdef LOCAL_ALLOC
    PAYMENT_DATA *PaymentDataPtr;
#endif

if((iRc = ThrTpInit(pECB)) <0)
{
    // This is bad
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessPayment Thread %d iTermId %d
Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
        fclose(fp);
    }
    sprintf(buf, "Payment ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());
    ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
}

memset(&Term.pClientData[iTermId].PaymentData, 0, sizeof(PAYMENT_DATA));

Term.pClientData[iTermId].PaymentData.w_id = Term.pClientData[iTermId].w_id;

if ( (iError=GetPaymentData(pECB->lpszQueryString,
&Term.pClientData[iTermId].PaymentData)) != ERR_SUCCESS )
{
    ErrorMessage(pECB, iError, ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
    return;
}

#ifdef LOCAL_ALLOC
if ((PaymentDataPtr = (PAYMENT_DATA *) tmalloc("CARRAY", NULL,
sizeof(PAYMENT_DATA))) == NULL)
{
    iRc = tperrno;
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessPayment Thread %d iTermId %d
tpalloc tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
        fclose(fp);
    }
    sprintf(buf, "Payment Tmalloc %d", iRc);
    ErrorMessage(pECB, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
    tfree((char *)PaymentDataPtr);
    return;
}

*PaymentDataPtr = Term.pClientData[iTermId].PaymentData;

ilen = sizeof(PAYMENT_DATA);
olen = &ilen;

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");

```

```

        fprintf(fp, "** ProcessPayment Thread %d iTermId %d
PaymentDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId, &PaymentDataPtr);
        fclose(fp);
    }

    if (tpcall("PAYMENT", (char *)PaymentDataPtr, ilen, (char **)&PaymentDataPtr,
(long *)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessPayment Thread %d iTermId %d
tpcall tperrno %d \r\n",
                    GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Payment tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
        tfree((char *)PaymentDataPtr);
        return;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessPayment Thread %d iTermId %d
PaymentDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId, &PaymentDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].PaymentData = *PaymentDataPtr;

    iRc = PaymentDataPtr->retval;

    tfree((char *)PaymentDataPtr);

#else

*Term.pClientData[iTermId].PaymentDataPtr =
Term.pClientData[iTermId].PaymentData;

ilen = sizeof(PAYMENT_DATA);
olen = &ilen;

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** ProcessPayment Thread %d iTermId %d
PaymentDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].PaymentDataPtr);
    fclose(fp);
}

if (tpcall("PAYMENT", (char *)Term.pClientData[iTermId].PaymentDataPtr, ilen,

```

```

        (char **) &Term.pClientData[iTermId].PaymentDataPtr, (long
*)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessPayment Thread %d iTermId %d
tpcall tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Payment tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
        tpfree((char *)PaymentDataPtr);
        return;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessPayment Thread %d iTermId %d
PaymentDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
            &Term.pClientData[iTermId].PaymentDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].PaymentData =
    *Term.pClientData[iTermId].PaymentDataPtr;

    iRc = Term.pClientData[iTermId].PaymentDataPtr->retval;

#endif
#ifdef BOIL
    Sleep( 2200 );
#endif

    if ( iRc == 0 )
        ErrorMessage(pECB, ERR_PAYMENT_INVALID_CUSTOMER, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
    else if ( iRc < 0 ) {
        ErrorMessage(pECB, ERR_PAYMENT_NOT_PROCESSED, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
    }
    else
        WriteZString(pECB, MakePaymentForm(iTermId, iSyncId, FALSE) );

    return;
}

/* FUNCTION: void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
int iSyncId)
*
* PURPOSE:      This function gets and validates the input data from the Order
Status
*
*               form filling in the required input variables. It then
calls the

```

```

        SQLOrderStatus transaction, constructs the output form
and writes it
        back to client browser.
*
* ARGUMENTS:      EXTENSION_CONTROL_BLOCK *pECB      passed in structure
pointer from inetsrv.
*
*               int
iTermId      client browser terminal id
*
*               int
iSyncId      client browser sync id
*
* RETURNS:        None
*
* COMMENTS:       None
*/

static void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId)
{
    int
    long ilen, *olen;
    // PECBINFO pEcbInfo;
    char buf[40];
#ifdef LOCAL_ALLOC
    ORDER_STATUS_DATA *OrderStatusDataPtr;
#endif

    if((iRc = ThrTpInit(pECB)) < 0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d
Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);
        }
        sprintf(buf, "OrderStatus ThrTpInit < 0 >> Thread %d",
            GetCurrentThreadId());
        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
    }

    memset(&Term.pClientData[iTermId].OrderStatusData, 0,
sizeof(ORDER_STATUS_DATA));

    Term.pClientData[iTermId].OrderStatusData.w_id =
Term.pClientData[iTermId].w_id;

    if ( (iError=GetOrderStatusData(pECB->lpszQueryString,
&Term.pClientData[iTermId].OrderStatusData)) != ERR_SUCCESS )
    {
        ErrorMessage(pECB, iError, ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
        return;
    }

#ifdef LOCAL_ALLOC
    if ((OrderStatusDataPtr = (ORDER_STATUS_DATA *) tmalloc("CARRAY", NULL,
sizeof(ORDER_STATUS_DATA))) == NULL)
    {

```

```

        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d\n",
                tpalloc tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "OrderStatus Tpcalloc %d", iRc);
        ErrorMessage(NULL, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
        tpfree((char *)OrderStatusDataPtr);
        return;
    }

    *OrderStatusDataPtr = Term.pClientData[iTermId].OrderStatusData;

    ilen = sizeof(ORDER_STATUS_DATA);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d\n",
            OrderStatusDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId, &OrderStatusDataPtr);
        fclose(fp);
    }

    if (tpcall("ORDERSTATUS", (char *)OrderStatusDataPtr, ilen, (char
        **)&OrderStatusDataPtr, (long *)olen, TPSIGRSTRT) == -1)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d\n",
                tpcall tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "OrderStatus tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
        tpfree((char *)OrderStatusDataPtr);
        return;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessOrderStatus Thread %d iTermId %d\n",
            OrderStatusDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId, &OrderStatusDataPtr);
        fclose(fp);
    }

    Term.pClientData[iTermId].OrderStatusData = *OrderStatusDataPtr;

```

```

        iRc = OrderStatusDataPtr->retval;

        tpfree((char *)OrderStatusDataPtr);

    #else

        *Term.pClientData[iTermId].OrderStatusDataPtr =
            Term.pClientData[iTermId].OrderStatusData;

        ilen = sizeof(ORDER_STATUS_DATA);
        olen = &ilen;

        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d\n",
                OrderStatusDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId,
                &Term.pClientData[iTermId].OrderStatusDataPtr);
            fclose(fp);
        }

        if (tpcall("ORDERSTATUS", (char *)Term.pClientData[iTermId].OrderStatusDataPtr,
            ilen,
            (char **)&Term.pClientData[iTermId].OrderStatusDataPtr, (long
                *)olen, TPSIGRSTRT) == -1)
        {
            iRc = tperrno;
            {
                FILE *fp;

                fp = fopen(szTpccLogPath, "ab");
                fprintf(fp, "** ProcessOrderStatus Thread %d iTermId %d\n",
                    tpcall tperrno %d \r\n",
                    GetCurrentThreadId(), iTermId, iRc);
                fclose(fp);
            }
            sprintf(buf, "OrderStatus tpcall %d", iRc);
            ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
            // tpfree((char *)Term.pClientData[iTermId].OrderStatusDataPtr);
            return;
        }

        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "*** ProcessOrderStatus Thread %d iTermId %d\n",
                OrderStatusDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId,
                &Term.pClientData[iTermId].OrderStatusDataPtr);
            fclose(fp);
        }

        Term.pClientData[iTermId].OrderStatusData =
            *Term.pClientData[iTermId].OrderStatusDataPtr;

        iRc = Term.pClientData[iTermId].OrderStatusDataPtr->retval;

```

```

#endif
#ifdef BOIL
Sleep( 3300 );
#endif

    if ( iRc == 0 )
        ErrorMessage(pECB, ERR_NOSUCH_CUSTOMER, ERR_TYPE_WEBDLL, NULL, iTermId,
iSyncId);
    else if ( iRc < 0 ) {
        ErrorMessage(pECB, ERR_ORDER_STATUS_NOT_PROCESSED, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
    }
    else
        WriteZString(pECB, MakeOrderStatusForm(iTermId, iSyncId, FALSE) );

    return;
}

/* FUNCTION: void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
int iSyncId)
*
* PURPOSE:      This function gets and validates the input data from the
delivery form
*               filling in the required input variables. It then calls
the PostDeliveryInfo
*               Api, The client is then informed that the transaction
has been posted.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB passed in structure
pointer from inetrv.
*
*               int
iTermId client browser terminal id
*               int
iSyncId clinet browser sync id
*
* RETURNS:     None
*
* COMMENTS:    None
*/

static void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId)
{
    int iRc;
    int iError;
    char szTmp[26];
    long ilen;
    int tmp;
    char buf[40];
#ifdef LOCAL_ALLOC
    DELIVERY_DATA *DeliveryDataPtr;
#endif

    if((iRc = ThrTpInit(pECB)) <0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");

```

```

                fprintf(fp, "** ProcessDelivery Thread %d iTermId %d
Failed ThrTpInit \r\n",
                    GetCurrentThreadId(), iTermId );
                fclose(fp);
            }
            sprintf(buf, "Delivery ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());
            ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
        }

        memset(&Term.pClientData[iTermId].DeliveryData, 0,
sizeof(DELIVERY_DATA));

        if ( !GetKeyValue(pECB->lpszQueryString, "OCD*", szTmp, sizeof(szTmp)) )
        {
            ErrorMessage(pECB, ERR_DELIVERY_MISSING_OCD_KEY,
ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
            return;
        }

        if ( !IsNumeric(szTmp) )
        {
            ErrorMessage(pECB, ERR_DELIVERY_CARRIER_INVALID,
ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
            return;
        }

        Term.pClientData[iTermId].DeliveryData.o_carrier_id = atoi(szTmp);
        tmp = Term.pClientData[iTermId].DeliveryData.o_carrier_id * 70;

        if ( Term.pClientData[iTermId].DeliveryData.o_carrier_id > 10 ||
Term.pClientData[iTermId].DeliveryData.o_carrier_id < 1 )
        {
            ErrorMessage(pECB, ERR_DELIVERY_CARRIER_ID_RANGE,
ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
            return;
        }

        Term.pClientData[iTermId].DeliveryData.w_id = Term.pClientData[iTermId].w_id;

        GetLocalTime(&Term.pClientData[iTermId].DeliveryData.queue_time);

// if ( (iError=GetDeliveryData(pECB->lpszQueryString,
&Term.pClientData[iTermId].DeliveryData)) != ERR_SUCCESS )
// {
//     ErrorMessage(pECB, iError, ERR_TYPE_WEBDLL, NULL, iTermId, iSyncId);
//     return;
// }

#ifdef LOCAL_ALLOC
        if ((DeliveryDataPtr = (DELIVERY_DATA *) tmalloc("CARRAY", NULL,
sizeof(DELIVERY_DATA))) == NULL)
        {
            iRc = tperrno;
            {
                FILE *fp;

                fp = fopen(szTpccLogPath, "ab");
                fprintf(fp, "** ProcessDelivery Thread %d iTermId %d
tpalloc tperrno %d \r\n",
                    GetCurrentThreadId(), iTermId, iRc);

```

```

        fclose(fp);
    }
    sprintf(buf, "Delivery Tpcalloc %d", iRc);
    ErrorMessage(NULL, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
    tpfree((char *)DeliveryDataPtr);
    return;
}

*DeliveryDataPtr = Term.pClientData[iTermId].DeliveryData;

ilen = sizeof(DELIVERY_DATA);

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** ProcessDelivery Thread %d iTermId %d
DeliveryDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId, &DeliveryDataPtr);
    fclose(fp);
}

if (tpacall("DELIVERY", (char *)DeliveryDataPtr, ilen, TPNOREPLY|TPSIGRSTRT) !=
0) {
    iRc = tpererrno;
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessDelivery Thread %d iTermId %d
tpcall tpererrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
        fclose(fp);

        sprintf(buf, "Delivery tpcall %d", iRc);
        ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
        tpfree((char *)DeliveryDataPtr);
        return;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "*** ProcessDelivery Thread %d iTermId %d
DeliveryDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId, &DeliveryDataPtr);
        fclose(fp);
    }

    tpfree((char *)DeliveryDataPtr);
}

#else

*Term.pClientData[iTermId].DeliveryDataPtr =
Term.pClientData[iTermId].DeliveryData;

ilen = sizeof(DELIVERY_DATA);
olen = &ilen;

if ( dLog )

```

```

    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessDelivery Thread %d iTermId %d
DeliveryDataPtr: %x \r\n",
                GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].DeliveryDataPtr);
        fclose(fp);
    }

    if (tpcall("DELIVERY", (char *)Term.pClientData[iTermId].DeliveryDataPtr, ilen,
(char *)&Term.pClientData[iTermId].DeliveryDataPtr, (long
*)olen, TPSIGRSTRT) == -1) {
        iRc = tpererrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessDelivery Thread %d iTermId %d
tpcall tpererrno %d \r\n",
                    GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);

            sprintf(buf, "Delivery tpcall %d", iRc);
            ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
            // tpfree((char *)Term.pClientData[iTermId].DeliveryDataPtr);
            return;
        }

        if ( dLog )
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "*** ProcessDelivery Thread %d iTermId %d
DeliveryDataPtr: %x \r\n",
                    GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].DeliveryDataPtr);
            fclose(fp);
        }

        iRc = Term.pClientData[iTermId].DeliveryDataPtr->retval;
    }

#endif

#ifdef TSLICE
    Sleep(tmp); // Relinquish thread timeslice
#endif

strcpy(Term.pClientData[iTermId].DeliveryData.execution_status, "Delivery
has been queued.");

#ifdef BOIL
    Sleep( 4400 );
#endif

    WriteZString(pECB, MakeDeliveryForm(iTermId, iSyncId, FALSE) );

    return;
}

```

```

/* FUNCTION: void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId,
int iSyncId)
*
* PURPOSE:      This function gets and validates the input data from the Stock
Level
*              form filling in the required input variables. It then
calls the
*              SQLStockLevel transaction, constructs the output form
and writes it
*              back to client browser.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB   passed in structure
pointer from inetsrv.
*              int
*              iTermId   client browser terminal id
*              int
*              iSyncId   client browser sync id
*
* RETURNS:     None
*
* COMMENTS:    None
*/

static void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK *pECB, int iTermId, int
iSyncId)
{
    int          iRc;
    int          iError;
    char         szTmp[26];
    BOOL         bRc;
    long         ilen, *olen;
    PECBINFO     pEcbInfo;
    char         buf[40];
#ifdef LOCAL_ALLOC
    STOCK_LEVEL_DATA *StockLevelDataPtr;
#endif

    if((iRc = ThrTpInit(pECB)) < 0)
    {
        // This is bad
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessStockLevel Thread %d iTermId %d\n",
Failed ThrTpInit \r\n",
                GetCurrentThreadId(), iTermId );
            fclose(fp);
        }
        sprintf(buf, "StockLevel ThrTpInit < 0 >> Thread %d",
GetCurrentThreadId());
        ErrorMessage(pECB, ERR_TPINIT_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
    }

    memset(&Term.pClientData[iTermId].StockLevelData, 0, sizeof(STOCK_LEVEL_DATA));
    Term.pClientData[iTermId].StockLevelData.w_id = Term.pClientData[iTermId].w_id;
    Term.pClientData[iTermId].StockLevelData.d_id = Term.pClientData[iTermId].d_id;

    if ( !GetKeyValue(pECB->lpszQueryString, "TT*", szTmp, sizeof(szTmp)) )
    {

```

```

        ErrorMessage(pECB, ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
        return;
    }

    if ( !IsNumeric(szTmp) )
    {
        ErrorMessage(pECB, ERR_STOCKLEVEL_THRESHOLD_INVALID, ERR_TYPE_WEBDLL,
NULL, iTermId, iSyncId);
        return;
    }

    Term.pClientData[iTermId].StockLevelData.thresh_hold = atoi(szTmp);

    if ( Term.pClientData[iTermId].StockLevelData.thresh_hold >= 100 ||
Term.pClientData[iTermId].StockLevelData.thresh_hold < 0 )
    {
        ErrorMessage(pECB, ERR_STOCKLEVEL_THRESHOLD_RANGE, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
        return;
    }

#ifdef LOCAL_ALLOC
    if ((StockLevelDataPtr = (STOCK_LEVEL_DATA *) tmalloc("CARRAY", NULL,
sizeof(STOCK_LEVEL_DATA))) == NULL)
    {
        iRc = tperrno;
        {
            FILE *fp;

            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ProcessStockLevel Thread %d iTermId %d\n",
tpalloc tperrno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
            fclose(fp);
        }
        sprintf(buf, "Tmalloc %d", iRc);
        ErrorMessage(NULL, ERR_TPALLOC_BAD, ERR_TYPE_TUXEDO, buf, 0, 0);
        tpfree((char *)StockLevelDataPtr);
        return;
    }

    *StockLevelDataPtr = Term.pClientData[iTermId].StockLevelData;

    ilen = sizeof(STOCK_LEVEL_DATA);
    olen = &ilen;

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessStockLevel Thread %d iTermId %d\n",
StockLevelDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId, &StockLevelDataPtr);
        fclose(fp);
    }

    if (tpcall("STOCKLEVEL", (char *) StockLevelDataPtr, ilen, (char **)
&StockLevelDataPtr, (long *) olen, TPSIGRSTRT) == -1) {
        iRc = tperrno;
        {
            FILE *fp;

```

```

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessStockLevel Thread %d iTermId %d
tpcall tperno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
        fclose(fp);
    }
    sprintf(buf, "StockLevel tpcall %d", iRc);
    ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
    tfree((char *)StockLevelDataPtr);
    return;
}

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "*** ProcessStockLevel Thread %d iTermId %d
StockLevelDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId, &StockLevelDataPtr);
    fclose(fp);

    Term.pClientData[iTermId].StockLevelData = *StockLevelDataPtr;

    iRc = StockLevelDataPtr->retval;

    tfree((char *)StockLevelDataPtr);
}

#else

*Term.pClientData[iTermId].StockLevelDataPtr =
Term.pClientData[iTermId].StockLevelData;

ilen = sizeof(STOCK_LEVEL_DATA);
olen = &ilen;

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "** ProcessStockLevel Thread %d iTermId %d
StockLevelDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].StockLevelDataPtr);
    fclose(fp);
}

if (tpcall("STOCKLEVEL", (char *)Term.pClientData[iTermId].StockLevelDataPtr,
ilen,
        (char **)&Term.pClientData[iTermId].StockLevelDataPtr, (long *)
olen, TPSIGRSTRT) == -1) {
    iRc = tperno;
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "** ProcessStockLevel Thread %d iTermId %d
tpcall tperno %d \r\n",
                GetCurrentThreadId(), iTermId, iRc);
        fclose(fp);
    }
}

```

```

    }
    sprintf(buf, "StockLevel tpcall %d", iRc);
    ErrorMessage(pECB, ERR_TPCALL_BAD, ERR_TYPE_TUXEDO, buf, iTermId, iSyncId);
    tfree((char *)StockLevelDataPtr);
}
return;
}

if ( dLog )
{
    FILE *fp;

    fp = fopen(szTpccLogPath, "ab");
    fprintf(fp, "*** ProcessStockLevel Thread %d iTermId %d
StockLevelDataPtr: %x \r\n",
            GetCurrentThreadId(), iTermId,
&Term.pClientData[iTermId].StockLevelDataPtr);
    fclose(fp);

    Term.pClientData[iTermId].StockLevelData =
*Term.pClientData[iTermId].StockLevelDataPtr;

    iRc = Term.pClientData[iTermId].StockLevelDataPtr->retval;
}

#endif

#ifdef BOIL
    Sleep( 5500 );
#endif

if ( iRc == 0 ) {
    ErrorMessage(pECB, ERR_STOCKLEVEL_NOT_PROCESSED, ERR_TYPE_WEBDLL, NULL,
iTermId, iSyncId);
}
else
    WriteZString(pECB, MakeStockLevelForm(iTermId, iSyncId, FALSE) );

return;
}

/* FUNCTION: int GetNewOrderData(LPSTR lpszQueryString, NEW_ORDER_DATA
*pNewOrderData)
*
* PURPOSE: This function extracts and validates the new order form data
from an http command string.
*
* ARGUMENTS: LPSTR lpszQueryString
client browser http command string
NEW_ORDER_DATA *pNewOrderData
pointer to new order data structure
*
* RETURNS: int
error code indicating reason for failure
ERR_SUCCESS
new_order input data successfully parsed
*
* COMMENTS: None
*/

static int GetNewOrderData(LPSTR lpszQueryString, NEW_ORDER_DATA *pNewOrderData)

```

```

{
    char    szTmp[26];
    int     i;
    short   spw;
    short   items;
    BOOL    bCheck;
    char *  s;

    s=lpszQueryString;

    GetVal(s, "DID*", szTmp, sizeof(szTmp))
    if(! *s)
        return ERR_NEWORDER_FORM_MISSING_DID;

    if ( !IsNumeric(szTmp) )
        return ERR_NEWORDER_DISTRICT_INVALID;

    pNewOrderData->d_id = atoi(szTmp);

    GetVal(s, "CID*", szTmp, sizeof(szTmp))
    if(! *s)
        return ERR_NEWORDER_CUSTOMER_KEY;

    if ( !IsNumeric(szTmp) )
        return ERR_NEWORDER_CUSTOMER_INVALID;
    pNewOrderData->c_id = atoi(szTmp);
    pNewOrderData->o_all_local=1;

    bCheck = FALSE;
    for(i=0, items=0; i<15; i++)
    {
        GetVal(s, szIID[i], szTmp, sizeof(szTmp))
        if(! *s)
            return ERR_NEWORDER_MISSING_IID_KEY;

        s=s-24;
        if ( szTmp[0] )
        {
            //if blank lines between item ids
            if ( bCheck )
                return ERR_NEWORDER_ITEM_BLANK_LINES;
            if ( !IsNumeric(szTmp) )
                return ERR_NEWORDER_ITEMID_INVALID;
            pNewOrderData->Ol[i].ol_i_id = atoi(szTmp);

            GetVal(s, szSP[i], szTmp, sizeof(szTmp))
            if(! *s)
                return ERR_NEWORDER_MISSING_SUPPW_KEY;
            if ( !IsNumeric(szTmp) )
                return ERR_NEWORDER_SUPPW_INVALID;

            if ( (spw = (short)atoi(szTmp)) <=0 || spw >
iMaxWareHouses )
                return ERR_NEWORDER_SUPPW_INVALID;

            if ((pNewOrderData->Ol[i].ol_supply_w_id = spw) !=
pNewOrderData->w_id)
                pNewOrderData->o_all_local=0;

            GetVal(s, szQty[i], szTmp, sizeof(szTmp))
            if(! *s)
                return ERR_NEWORDER_MISSING_QTY_KEY;

```

```

        if ( !IsNumeric(szTmp) )
            return ERR_NEWORDER_QTY_INVALID;

        pNewOrderData->Ol[i].ol_quantity = atoi(szTmp);
        items++;

        if ( pNewOrderData->Ol[i].ol_i_id >= 1000000 ||
pNewOrderData->Ol[i].ol_i_id < 1 )
            return ERR_NEWORDER_ITEMID_RANGE;
        if ( pNewOrderData->Ol[i].ol_quantity >= 100 ||
pNewOrderData->Ol[i].ol_quantity < 1 )
            return ERR_NEWORDER_QTY_RANGE;
    }
    else
    {
        GetVal(s, szSP[i], szTmp, sizeof(szTmp))
        if(! *s)
            return ERR_NEWORDER_MISSING_QTY_KEY;

        if ( szTmp[0] )
            return ERR_NEWORDER_SUPPW_WITHOUT_ITEMID;

        GetVal(s, szQty[i], szTmp, sizeof(szTmp))
        if(! *s)
            return ERR_NEWORDER_MISSING_QTY_KEY;

        if ( szTmp[0] )
            return ERR_NEWORDER_QTY_WITHOUT_ITEMID;

        bCheck = TRUE;
    }
}
if ( items == 0 )
    return ERR_NEWORDER_NOITEMS_ENTERED;

pNewOrderData->o_ol_cnt = items;

return ERR_SUCCESS;
}

/* FUNCTION: int GetPaymentData(LPSTR lpszQueryString, PAYMENT_DATA *pPaymentData)
*
* PURPOSE: This function extracts and validates the payment form data from
an http command string.
*
* ARGUMENTS: LPSTR lpszQueryString
client browser http command string
PAYMENT_DATA *pPaymentData
pointer to payment data structure
*
* RETURNS: int
error code indicating reason for failure
ERR_SUCCESS
all input data successfully parsed
*
* COMMENTS: None
*/

static int GetPaymentData(LPSTR lpszQueryString, PAYMENT_DATA *pPaymentData)
{
    char    szTmp[26];

```



```

char      *ptr;
char      *s;

s=lpszQueryString;

GetVal(s, "DID*", szTmp, sizeof(szTmp))
if(! *s)
    return ERR_PAYMENT_MISSING_DID_KEY;

if ( !IsNumeric(szTmp) )
    return ERR_PAYMENT_DISTRICT_INVALID;
pPaymentData->d_id = atoi(szTmp);

GetVal(s, "CID*", szTmp, sizeof(szTmp))
if(! *s)
    return ERR_PAYMENT_MISSING_CID_KEY;

if ( szTmp[0] && !IsNumeric(szTmp) )
    return ERR_PAYMENT_CUSTOMER_INVALID;

pPaymentData->c_id = atoi(szTmp);

if ( szTmp[0] == 0 )
{
    GetVal(s, "CLT*", szTmp, sizeof(szTmp))
    if(! *s)
        return ERR_PAYMENT_MISSING_CLT;
    _strupr( szTmp );

    strcpy(pPaymentData->c_last, szTmp);
    if ( strlen(pPaymentData->c_last) > 16 )
        return ERR_PAYMENT_LAST_NAME_TO_LONG;
}
else
{
    GetVal(s, "CLT*", szTmp, sizeof(szTmp))
    if(! *s)
        return ERR_PAYMENT_MISSING_CLT_KEY;
    if ( szTmp[0] )
        return ERR_PAYMENT_CID_AND_CLT;
}

s=lpszQueryString;

GetVal(s, "CWI*", szTmp, sizeof(szTmp))
if(! *s)
    return ERR_PAYMENT_MISSING_CWI_KEY;

if ( !IsNumeric(szTmp) )
    return ERR_PAYMENT_CWI_INVALID;

pPaymentData->c_w_id = atoi(szTmp);

GetVal(s, "CDI*", szTmp, sizeof(szTmp))
if(! *s)
    return ERR_PAYMENT_MISSING_CDI_KEY;
if ( !IsNumeric(szTmp) )
    return ERR_PAYMENT_CDI_INVALID;
pPaymentData->c_d_id = atoi(szTmp);

GetVal(s, "HAM*", szTmp, sizeof(szTmp))

```

```

if(! *s)
    return ERR_PAYMENT_MISSING_HAM_KEY;

ptr = szTmp;

while( *ptr )
{
    if ( *ptr == '.' )
    {
        ptr++;
        if ( !*ptr )
            break;
        if ( *ptr < '0' || *ptr > '9' )
            return ERR_PAYMENT_HAM_INVALID;
        ptr++;
        if ( !*ptr )
            break;
        if ( *ptr < '0' || *ptr > '9' )
            return ERR_PAYMENT_HAM_INVALID;
        if ( !*ptr )
            return ERR_PAYMENT_HAM_INVALID;
    }
    else if ( *ptr < '0' || *ptr > '9' )
        return ERR_PAYMENT_HAM_INVALID;
    ptr++;
}

pPaymentData->h_amount = atof(szTmp);
if ( pPaymentData->h_amount >= 10000.00 || pPaymentData->h_amount < 0 )
    return ERR_PAYMENT_HAM_RANGE;

return ERR_SUCCESS;
}

/* FUNCTION: int GetOrderStatusData(LPSTR lpszQueryString, ORDER_STATUS_DATA
*pOrderStatusData)
*
* PURPOSE:      This function extracts and validates the payment form data from
an http command string.
*
* ARGUMENTS:    LPSTR                                lpszQueryString
                client browser http command string
                ORDER_STATUS_DATA *pOrderStatusData
                pointer to order status data structure
*
* RETURNS:      int
                error code indicating reason for failure
                ERR_SUCCESS
                successfully parsed all required input data
*
* COMMENTS:     None
*
*/
static int GetOrderStatusData(LPSTR lpszQueryString, ORDER_STATUS_DATA
*pOrderStatusData)
{
    char      szTmp[26];
    char      *s;

    s=lpszQueryString;

    GetVal(s, "DID*", szTmp, sizeof(szTmp))
    if(! *s)

```

```

        return ERR_ORDERSTATUS_MISSING_DID_KEY;
    if ( !IsNumeric(szTmp) )
        return ERR_ORDERSTATUS_DID_INVALID;
    pOrderStatusData->d_id = atoi(szTmp);

    GetVal(s, "CID*", szTmp, sizeof(szTmp))
    if(! *s)
        return ERR_ORDERSTATUS_MISSING_CID_KEY;

    if ( szTmp[0] == 0 )
    {
        pOrderStatusData->c_id = 0;
        GetVal(s, "CLT*", szTmp, sizeof(szTmp))
        if(! *s)
            return ERR_ORDERSTATUS_MISSING_CLT_KEY;
        _strupr( szTmp );
        strcpy(pOrderStatusData->c_last, szTmp);
        if ( strlen(pOrderStatusData->c_last) > 16 )
            return ERR_ORDERSTATUS_CLT_RANGE;
    }
    else
    {
        if ( !IsNumeric(szTmp) )
            return ERR_ORDERSTATUS_CID_INVALID;
        pOrderStatusData->c_id = atoi(szTmp);
        GetVal(s, "CLT*", szTmp, sizeof(szTmp))
        if(! *s)
            return ERR_ORDERSTATUS_MISSING_CLT_KEY;
        if ( szTmp[0] )
            return ERR_ORDERSTATUS_CID_AND_CLT;
    }

    return ERR_SUCCESS;
}

/* FUNCTION: BOOL ReadRegistrySettings(void)
 *
 * PURPOSE:      This function reads the NT registry for startup parameters.
 * There parameters are
 *               under the TPCC key.
 *
 * ARGUMENTS:   None
 *
 * RETURNS:     None
 *
 * COMMENTS:    This function also sets up required operation variables to their
 * default value
 *               so if registry is not setup the default
 * values will be used.
 */

static BOOL ReadRegistrySettings(void)
{
    HKEY    hKey;
    DWORD   size;
    DWORD   type;
    char    szTmp[256];

    bLog    = FALSE;
    dLog    = FALSE;
    iMaxWareHouses = 0;
    iThreads = 5;

```

```

    iDelayMs      = 100;
    iDeadlockRetry = (short)3;
    strcpy(szTpccLogPath, "tpcclog.");

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE, "SOFTWARE\\Microsoft\\TPCC", 0,
    KEY_READ, &hKey) != ERROR_SUCCESS )
        return TRUE;
    size = sizeof(szTmp);

    if ( RegQueryValueEx(hKey, "PATH", 0, &type, szTmp, &size) ==
    ERROR_SUCCESS )
    {
        strcpy(szTpccLogPath, szTmp);
        strcat(szTpccLogPath, "tpcclog.");
        strcpy(szErrorLogPath, szTmp);
        strcat(szErrorLogPath, "tpccerr.");
    }

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "LOG", 0, &type, szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !stricmp(szTmp, "ON") )
            bLog = TRUE;
    }

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "DEBUG", 0, &type, szTmp, &size) ==
    ERROR_SUCCESS )
    {
        if ( !stricmp(szTmp, "ON") )
            dLog = TRUE;
    }

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "MaximumWarehouses", 0, &type, szTmp, &size) ==
    ERROR_SUCCESS )
    {
        iMaxWareHouses = atoi(szTmp);
        if ( iMaxWareHouses == 0 )
            iMaxWareHouses = 0;
    }

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "NumberOfDeliveryThreads", 0, &type, szTmp,
    &size) == ERROR_SUCCESS )
        iThreads = atoi(szTmp);
    if ( !iThreads )
        iThreads = 5;

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "BackoffDelay", 0, &type, szTmp, &size) ==
    ERROR_SUCCESS )
        iDelayMs = atoi(szTmp);
    if ( !iDelayMs )
        iDelayMs = 100;

    size = sizeof(szTmp);
    if ( RegQueryValueEx(hKey, "DeadlockRetry", 0, &type, szTmp, &size) ==
    ERROR_SUCCESS )
        iDeadlockRetry = (short)atoi(szTmp);
    if ( !iDeadlockRetry )

```

```

        iDeadlockRetry = (short)3;

        size = sizeof(szTmp);
        if ( RegQueryValueEx(hKey, "MaxConnections", 0, &type, szTmp, &size) ==
ERROR_SUCCESS )
            iMaxConnections = (short)atoi(szTmp);
            if ( !iMaxConnections )
                iMaxConnections = (short)25;

        RegCloseKey(hKey);

        return FALSE;
}

/* FUNCTION: BOOL IsNumeric(char *ptr)
 *
 * PURPOSE:      This function determines if a string is numeric. It fails if any
characters other
 *               than numeric and null terminator are present.
 *
 * ARGUMENTS:   char          *ptr      pointer to string to
check.
 *
 * RETURNS:     BOOL          FALSE    if string is not all numeric
 *               TRUE         if string
contains only numeric characters i.e. '0' - '9'
 *
 * COMMENTS:    None
 *
 */

static BOOL IsNumeric(char *ptr)
{
    if ( *ptr == 0 )
        return FALSE;

    while( *ptr && isdigit(*ptr) )
        ptr++;
    return ( !*ptr );
}

#ifdef USE_ODBC
/* FUNCTION: static int ODBCError(DBPROCESS *dbproc)
 *
 * PURPOSE:      This function Handles the processing of errors from
ODBC APIs
 *
 * ARGUMENTS:   PDBPROCESS
 *
 * RETURNS:     BOOL          FALSE    if string is not all
numeric
 *               TRUE         if
string contains only numeric characters i.e. '0' - '9'
 *
 * COMMENTS:    None
 *
 */

static int ODBCError(DBPROCESS *dbproc)
{

```

```

RETCODE      rc;
SDWORD       lNativeError;
BOOL         bError;
char         szState[6];
char         szMsg[SQL_MAX_MESSAGE_LENGTH];
char         timebuf[128];
char         datebuf[128];

pdbproc->deadlock_detected = TRUE;
bError = FALSE;

while( SQLError(dbproc->henv,

                                dbproc->hdbc,
                                dbproc->hstmt,
                                szState,
                                &lNativeError,
                                szMsg,
                                sizeof(szMsg),
                                NULL) !=

SQL_NO_DATA_FOUND )
{
    if (lNativeError == 1205)
        dbproc->deadlock_detected = TRUE;
    else
    {
        _strtime(timebuf);
        _strdate(datebuf);

        h_printf(dbproc->pECB, "%s %s : ODBC Error:
State=%s, Error=%ld, %s\n", datebuf, timebuf, szState, lNativeError, szMsg);
        bError = TRUE;
    }
    if ( bError )
        return -1;
    return dbproc->deadlock_detected;
}
#endif

/* FUNCTION: void FormatHTMLString(char *szBuff, int iLen, char *szStr)
 *
 * PURPOSE:      This function Handles translation of HTML specific character
field data
 *               when an HTML output form is generated.
 *
 * ARGUMENTS:   char          *szBuff  Returned string information
char          *szStr    input string to be
formatted.
 *
 * RETURNS:     int           iLen     Length of
returned string
 *
 * RETURNS:     none
 *
 * COMMENTS:    The length paramter is the absolute length of the returned
string in
 *               HTML characters. For example the input
string > would be returned as
 *               &gt; which would be counted as 1
character.If the number of input
 *               characters is less than the iLen parameter
spaces are appended to
 *               the end of the string to ensure that at
least iLen characters are

```

```

*                               returned in the szBuff parameter.
*
*/
static void FormatHTMLString(char *szBuff, char *szStr, int iLen)
{
    while( iLen && *szStr )
    {
        switch( *szStr )
        {
            case '>':
                *szBuff++ = '&';
                *szBuff++ = 'g';
                *szBuff++ = 't';
                *szBuff++ = ';';
                szStr++;
                break;

            case '<':
                *szBuff++ = '&';
                *szBuff++ = 'l';
                *szBuff++ = 't';
                *szBuff++ = ';';
                szStr++;
                break;

            case '&':
                *szBuff++ = '&';
                *szBuff++ = 'a';
                *szBuff++ = 'm';
                *szBuff++ = 'p';
                *szBuff++ = ';';
                szStr++;
                break;

            case '\\':
                *szBuff++ = '&';
                *szBuff++ = 'q';
                *szBuff++ = 'u';
                *szBuff++ = 'o';
                *szBuff++ = 't';
                *szBuff++ = ';';
                szStr++;
                break;

            default:
                *szBuff++ = *szStr++;
                break;
        }
        iLen--;
    }
    while( iLen-- )
        *szBuff++ = ' ';

    *szBuff = 0;
    return;
}

static int ThrTpInit(EXTENSION_CONTROL_BLOCK *pECB)
{
    // use CS for ttpalloc ?
    static int num_tpinit=0;
    static int x=1;
    static int once=0;
    static CRITICAL_SECTION    TpCriticalSection;

```

```

int iRc, TpRc, lasterr;
char buf[40];
int retry = 0;
BOOL Success = FALSE;

if(!TlsGetValue(TLSIsTpInitedKey))
{
    if (!once)
    {
        InitializeCriticalSection(&TpCriticalSection);
        once=1;
    }

    if ( dLog )
    {
        FILE *fp;

        fp = fopen(szTpccLogPath, "ab");
        fprintf(fp, "%s In ThrTpinit Thread %d * \r\n",
GetCurrentThreadId());
        fclose(fp);
    }

    while ( retry <= TP_MAX_RETRIES )
    {
        EnterCriticalSection(&TpCriticalSection);

        if(tpinf == NULL)
        {
            if ((tpinf = ( TPINIT *)tpalloc("TPINIT", NULL,
sizeof(TPINIT))) == NULL)
            {
                TpRc = tperrno;
                {
                    FILE *fp;

                    fp = fopen(szTpccLogPath, "ab");
                    fprintf(fp, ">>>> ThrTpInit %d :
tpalloc of tpinit failed: %d \r\n",
GetCurrentThreadId(),
TpRc);

                    fclose(fp);
                }
                tpinf = NULL;
                retry++;
            }
            LeaveCriticalSection(&TpCriticalSection);
            Sleep(50); // Relinquish thread timeslice
            continue;
        }
        tpinf->flags|=TPMULTICONTEXTS;
    }

    // Do the TPINIT

    itoa(++num_tpinit, tpinf->cltname, 10);
    iRc = tpinit(tpinf);

    // check tpinit() ?
    if (iRc < 0 )
    {
        lasterr = GetLastError();
    }
}

```

```

        retry++ ;
        if ( dLog )
        {
            FILE *fp;
            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ThrTpInit Thread %d
Retry : count %d * \r\n",
            GetCurrentThreadId(),
            retry);
            fclose(fp);
        }
        else
        {
            Success = TRUE;
            tpfree( ( char * ) tpinf);
            tpinf=NULL;
            break;
        }
        --num_tpinits;
        LeaveCriticalSection(&TpCriticalSection);
        Sleep(50); // Relinquish thread
timeslice
    } // retry the tpinit if it failed the first time
    LeaveCriticalSection(&TpCriticalSection);
    if ( Success == FALSE )
    {
        TpRc = tperrno;
        {
            FILE *fp;
            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> ThrTpInit Thread %d Failed
tperrno= %d Last Err= %d \r\n",
            GetCurrentThreadId(), TpRc,
            lasterr);
            fclose(fp);
        }
        sprintf(buf,"Thread %d: In ThrTpInit iRc=%d
Tperrno=%d",
            GetCurrentThreadId(),iRc, TpRc);
        ErrorMessage(pECB, ERR_TPINIT_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
        return -1;
    }
    if ( Success == TRUE )
    {
        if ( dLog )
        {
            FILE *fp;
            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ThrTpInit Thread %d Success
retry count %d * \r\n",
            GetCurrentThreadId(), retry);
            fclose(fp);
        }
    }

```

```

        if ( ( iRc=TlsSetValue(TLSIsTpInitedKey,&x) == 0)
        {
            FILE *fp;
            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, ">>>> ThrTpInit %d :
TlsSetValue Failed iRc: %d \r\n",
            GetCurrentThreadId(), iRc);
            fclose(fp);
        }
        sprintf(buf,"TlsSetValue %d",iRc);
        ErrorMessage(pECB, ERR_TPINIT_BAD,
ERR_TYPE_TUXEDO, buf, 0, 0);
        return -1;
    }
    else
    {
        if ( dLog )
        {
            FILE *fp;
            fp = fopen(szTpccLogPath, "ab");
            fprintf(fp, "** ThrTpInit Thread %d already tpinited *
\r\n", GetCurrentThreadId());
            fclose(fp);
        }
    }
    return 0;
}

```

tpcc.def

LIBRARY TPCC.DLL

EXPORTS

```

    GetExtensionVersion @1
    HttpExtensionProc @2

```

tpcc.h

```

// File: TPCC.H
// Microsoft TPC-C Kit Ver. 4.00
// Copyright Microsoft, 1996, 1997, 1998
// Purpose: Header file for TPC-C database loader

```

```

// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.00"

```

```

// General headers
#include <windows.h>
#include <winbase.h>
#include <stdlib.h>
#include <stdio.h>

```

```

#include <process.h>
#include <stddef.h>
#include <stdarg.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <sys\types.h>

// ODBC headers
#include <sql.h>
#include <sqlext.h>
#include <odbc.h>

// General constants
#define MILLI 1000
#define FALSE 0
#define TRUE 1
#define UNDEF -1
#define MINPRINTASCII 32
#define MAXPRINTASCII 126

// Default environment constants
#define SERVER ""
#define DATABASE "tpcc"
#define USER "sa"
#define PASSWORD ""

// Default loader arguments
#define BATCH 10000
#define DEFLOADPACKSIZE 32768
#define ORDERS_PER_DIST 3000
#define LOADER_RES_FILE "logs\\load.out"
#define LOADER_NURAND_C 123
#define DEF_STARTING_WAREHOUSE 1
#define BUILD_INDEX 1 // build both
// data and indexes
#define INDEX_ORDER 1 // build
// indexes before load
#define SCALE_DOWN 0 // build a normal
// scale database
#define INDEX_SCRIPT_PATH "scripts"

typedef struct
{
    char *server;
    char *database;
    char *user;
    char *password;
    BOOL tables_all;
    // set if loading all tables
    BOOL table_item;
    // set if loading ITEM table specifically
    BOOL table_warehouse; // set if
// loading WAREHOUSE, DISTRICT, and STOCK
    BOOL table_customer; //
// set if loading CUSTOMER and HISTORY
    BOOL table_orders; //
// set if loading NEW-ORDER, ORDERS, ORDER-LINE
    long num_warehouses;
    long batch;
    long verbose;
    long pack_size;
    long char *loader_res_file;
}

```

```

char *synch_servername;
long case_sensitivity;
long starting_warehouse;
long build_index;
long index_order;
long scale_down;
char *index_script_path;
} TPCCLDR_ARGS;

// String length constants
#define SERVER_NAME_LEN 20
#define DATABASE_NAME_LEN 20
#define USER_NAME_LEN 20
#define PASSWORD_LEN 20
#define TABLE_NAME_LEN 20
#define I_DATA_LEN 50
#define I_NAME_LEN 24
#define BRAND_LEN 1
#define LAST_NAME_LEN 16
#define W_NAME_LEN 10
#define ADDRESS_LEN 20
#define STATE_LEN 2
#define ZIP_LEN 9
#define S_DIST_LEN 24
#define S_DATA_LEN 50
#define D_NAME_LEN 10
#define FIRST_NAME_LEN 16
#define MIDDLE_NAME_LEN 2
#define PHONE_LEN 16
#define CREDIT_LEN 2
#define C_DATA_LEN 500
#define H_DATA_LEN 24
#define DIST_INFO_LEN 24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN 25
#define OL_DIST_INFO_LEN 24
#define C SINCE_LEN 23
#define H DATE_LEN 23
#define OL_DELIVERY_D_LEN 23
#define O_ENTRY_D_LEN 23

// Functions in random.c
void seed();
long irand();
double drand();
void WUCreate();
short WURand();
long RandomNumber(long lower, long upper);

// Functions in getargs.c;
void GetArgsLoader();
void GetArgsLoaderUsage();

// Functions in time.c
long TimeNow();

// Functions in strings.c
void MakeAddress();
void LastName();
int MakeAlphaString();
int MakeOriginalAlphaString();

```

```

int     MakeNumberString();
int     MakeZipNumberString();
void    InitString();
void    InitAddress();
void    PaddString();

```

tpcc.rc

```

//Microsoft Developer Studio generated resource script.
//
#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 2 resource.
//
#include "afxres.h"

////////////////////////////////////
#undef APSTUDIO_READONLY_SYMBOLS

////////////////////////////////////
// English (U.S.) resources

#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifdef WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // WIN32

#ifndef _MAC
////////////////////////////////////
//
// Version
//

VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,3,0,2
PRODUCTVERSION 0,3,0,2
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x40004L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "Comments", "TPC-C HTML DLL Server\0"
            VALUE "CompanyName", "Microsoft\0"
            VALUE "FileDescription", "tpcc\0"
            VALUE "FileVersion", "0, 3, 0, 2\0"
            VALUE "InternalName", "tpcc\0"
            VALUE "LegalCopyright", "Copyright © 1996\0"
            VALUE "OriginalFilename", "tpcc.dll\0"
            VALUE "ProductName", "Microsoft tpcc\0"

```

```

        VALUE "ProductVersion", "0, 3, 0, 2\0"
    END
END
BLOCK "VarFileInfo"
BEGIN
    VALUE "Translation", 0x409, 1200
END
END

#endif // !_MAC

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// TEXTINCLUDE
//

1 TEXTINCLUDE DISCARDABLE
BEGIN
    "resource.h\0"
END

2 TEXTINCLUDE DISCARDABLE
BEGIN
    "#include \"afxres.h\"\r\n"
    "\0"
END

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "\r\n"
    "\0"
END

#endif // APSTUDIO_INVOKED

#endif // English (U.S.) resources
////////////////////////////////////

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//

////////////////////////////////////
#endif // not APSTUDIO_INVOKED

```

trans.h

```

/*     FILE:                TRANS.H
 *
 *     Microsoft TPC-C Kit Ver. 3.00.000
 *     Audited 08/23/96   By Francois Raab
 *
 *     PURPOSE: Header file for ISAPI TPCC.DLL, defines structures and functions
 *     used in the isapi tpcc.dll.
 *
 */

```

```

*                               Copyright Microsoft inc. 1996, All Rights
Reserved
*
*   Author:      PhilipDu, from tpcc.h by DamienL
*               DamienL@Microsoft.com
*               philipdu@Microsoft.com
*/

#ifndef _INC_TRANS
#define _INC_TRANS

#ifdef USE_ODBC
#ifdef TIMESTAMP_STRUCT
#include <sqltypes.h>
#else
#ifdef _INC_SQLFRONT
#include <sqlfront.h>
#endif
#endif
#endif

#ifdef DBINT
typedef long DBINT;
#endif

#define DEFCLPACKSIZE      4096
#define DEADLOCKWAIT      10

// String length constants
#define SERVER_NAME_LEN    20
#define DATABASE_NAME_LEN 20
#define USER_NAME_LEN     20
#define PASSWORD_LEN      20
#define TABLE_NAME_LEN   20
#define I_DATA_LEN        50
#define I_NAME_LEN        24
#define BRAND_LEN         1
#define LAST_NAME_LEN     16
#define W_NAME_LEN        10
#define ADDRESS_LEN       20
#define STATE_LEN         2
#define ZIP_LEN           9
#define S_DIST_LEN        24
#define S_DATA_LEN        50
#define D_NAME_LEN        10
#define FIRST_NAME_LEN    16
#define MIDDLE_NAME_LEN   2
#define PHONE_LEN         16
#define DATETIME_LEN      30
#define CREDIT_LEN        2
#define C_DATA_LEN        250
#define H_DATA_LEN        24
#define DIST_INFO_LEN     24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN        25
#define OL_DIST_INFO_LEN 24

// transaction structures
typedef struct
{

```

```

short
long
char
short
char
ol_brand_generic[BRAND_LEN+1];
double
double
short
short
} OL_NEW_ORDER_DATA;

typedef struct
{
short
short
long
short
char
char
double
double
double
long
short
#ifdef USE_ODBC
TIMESTAMP_STRUCT
#else
DBDATEREC
#endif
short
double
long
int
char
execution_status[STATUS_LEN];
OL_NEW_ORDER_DATA
} NEW_ORDER_DATA;

typedef struct
{
short
short
long
short
short
double
#ifdef USE_ODBC
TIMESTAMP_STRUCT
#else
DBDATEREC
#endif
char
w_street_1[ADDRESS_LEN+1];
char
w_street_2[ADDRESS_LEN+1];
char
char
char
char
d_street_1[ADDRESS_LEN+1];
char
d_street_2[ADDRESS_LEN+1];
short
long
char
char
double
double
long
short
o_id;
o_commit_flag;
TIMESTAMP_STRUCT
o_entry_d;
DBDATEREC
o_entry_d;
short
double
long
int
char
OL_NEW_ORDER_DATA
ol[MAX_OL_NEW_ORDER_ITEMS];
short
short
long
short
short
double
TIMESTAMP_STRUCT
h_date;
DBDATEREC
h_date;
char
char
char
char
char
w_city[ADDRESS_LEN+1];
w_state[STATE_LEN+1];
w_zip[ZIP_LEN+1];
ol_supply_w_id;
ol_i_id;
ol_i_name[I_NAME_LEN+1];
ol_quantity;
ol_i_price;
ol_amount;
ol_stock;
num_warehouses;

```



```

char                d_city[ADDRESS_LEN+1];
char                d_state[STATE_LEN+1];
char                d_zip[ZIP_LEN+1];
char
char
c_first[FIRST_NAME_LEN+1];
char
c_middle[MIDDLE_NAME_LEN + 1];
char
char                c_last[LAST_NAME_LEN+1];
char
char
c_street_1[ADDRESS_LEN+1];
char
c_street_2[ADDRESS_LEN+1];
char
char                c_city[ADDRESS_LEN+1];
char                c_state[STATE_LEN+1];
char                c_zip[ZIP_LEN+1];
char                c_phone[PHONE_LEN+1];
#ifdef USE_ODBC
    TIMESTAMP_STRUCT    c_since;
#else
    DBDATEREC          c_since;
#endif
char                c_credit[CREDIT_LEN+1];
double              c_credit_lim;
double              c_discount;
double              c_balance;
char                c_data[200+1];
long                num_deadlocks;
int                 retval;
char
execution_status[STATUS_LEN];
} PAYMENT_DATA;

typedef struct
{
    long                ol_i_id;
    short               ol_supply_w_id;
    short               ol_quantity;
    double              ol_amount;
#ifdef USE_ODBC
    TIMESTAMP_STRUCT    ol_delivery_d;
#else
    DBDATEREC          ol_delivery_d;
#endif
} OL_ORDER_STATUS_DATA;

typedef struct
{
    short              w_id;
    short              d_id;
    long               c_id;
    char               c_first[FIRST_NAME_LEN+1];
    char               c_middle[MIDDLE_NAME_LEN+1];
    char               c_last[LAST_NAME_LEN+1];
    double             c_balance;
    long               o_id;
#ifdef USE_ODBC
    TIMESTAMP_STRUCT    o_entry_d;
#else
    DBDATEREC          o_entry_d;
#endif
    short              o_carrier_id;

```

```

OL_ORDER_STATUS_DATA
olOrderStatusData[MAX_OL_ORDER_STATUS_ITEMS];
short               o_ol_cnt;
long                num_deadlocks;
int                 retval;
char
} ORDER_STATUS_DATA;

typedef struct
{
    long                o_id;
} DEL_ITEM;

typedef struct
{
    short              w_id;
    short              o_carrier_id;
    SYSTEMTIME         queue_time;
    long               num_deadlocks;
    long               o_id[10];
    int                 retval;
    char
} DELIVERY_DATA;

typedef struct
{
    short              w_id;
    short              d_id;
    short              thresh_hold;
    long               low_stock;
    long               num_deadlocks;
    int                 retval;
    char
    execution_status[STATUS_LEN];
} STOCK_LEVEL_DATA;

#endif

```

Appendix B: Database Design

The TPC-C database was created with the following Transact-SQL scripts:

dbopt1.sql

```
-- File:      DBOPT1.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Sets database options for data load
```

```
use master
go
```

```
exec sp_dboption tpcc,'select into/bulkcopy',true
exec sp_dboption tpcc,'trunc. log on chkpt.',true
go
```

```
use tpcc
go
```

```
checkpoint
go
```

dbopt2.sql

```
-- File:      DBOPT2.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.01
--           Copyright Microsoft, 1996
-- Purpose:   Resets database options after data load
```

```
use master
go
```

```
sp_dboption tpcc,'select ',false
go
```

```
sp_dboption tpcc,'trunc. ',false
go
```

```
use tpcc
go
```

```
checkpoint
```

```
go

sp_configure allow,1
go

reconfigure with override
go

/*                                     */
/* Set option values for user-defined indexes */
/*                                     */

sp_indexoption 'customer','AllowPageLocks',FALSE
go
sp_indexoption 'customer','AllowRowLocks',TRUE
go
sp_indexoption 'district','AllowPageLocks',FALSE
go
sp_indexoption 'warehouse','AllowPageLocks',FALSE
go
sp_indexoption 'stock','AllowPageLocks',FALSE
go
sp_indexoption 'stock','AllowRowLocks',TRUE
go
sp_indexoption 'order_line','AllowPageLocks',TRUE
go
sp_indexoption 'order_line','AllowRowLocks',FALSE
go
sp_indexoption 'orders','AllowPageLocks',TRUE
go
sp_indexoption 'orders','AllowRowLocks',FALSE
go
sp_indexoption 'new_order','AllowRowLocks',FALSE
go
sp_indexoption 'item','AllowRowLocks',FALSE
go
sp_indexoption 'item','AllowPageLocks',FALSE
go

Print ' '
Print '*****'
Print 'Pre-specified Locking Hierarchy:'
Print '  Lockflag = 0 ==> No pre-specified
hierarchy'
Print '  Lockflag = 1 ==> Lock at Page-level then
Table-level'
Print '  Lockflag = 2 ==> Lock at Row-level then
Table-level'
Print '  Lockflag = 3 ==> Lock at Table-level'
Print ' '

select name,lockflags
from sysindexes
where object_id("warehouse")=id or
       object_id("district")=id or
       object_id("customer")=id or
       object_id("stock")=id or
       object_id("orders")=id or
       object_id("order_line")=id or
       object_id("history")=id or
       object_id("new_order")=id or
       object_id("item")=id

order by lockflags asc
```

```
go

sp_configure allow,0
go

reconfigure with override
go

exec sp_dboption tpcc, 'auto update statistics',
FALSE
exec sp_dboption tpcc, 'auto create statistics',
FALSE
go

exec sp_tableoption "district","pintable",true
exec sp_tableoption "warehouse","pintable",true
exec sp_tableoption "new_order","pintable",true
exec sp_tableoption "item","pintable",true
go
```

delivery.sql

```
-- File:      DELIVERY.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates delivery transaction stored
procedure
```

```
use tpcc
go
```

```
if exists (select name from sysobjects where name =
"tpcc_delivery" )
drop procedure tpcc_delivery
```

```
go
```

```
create proc tpcc_delivery @w_id
smallint,
```

```
@o_carrier_id smallint
as
```

```
declare @d_id tinyint,
        @o_id int,
        @c_id int,
        @total numeric(12,2),
        @oid1 int,
        @oid2 int,
        @oid3 int,
        @oid4 int,
        @oid5 int,
        @oid6 int,
        @oid7 int,
        @oid8 int,
        @oid9 int,
        @oid10 int
```

```
select @d_id = 0
```

```

begin tran d

  while (@d_id < 10)
  begin

    select @d_id = @d_id + 1,
           @total = 0,
           @o_id = 0

           select top 1 @o_id = no_o_id
           from new_order (serializable)

  uplock)
  where no_w_id = @w_id and
        no_d_id = @d_id
        order by no_o_id asc

  if (@@rowcount > 0)
  begin

-- claim the order for this district

    delete new_order
    where no_w_id = @w_id and
          no_d_id = @d_id and
          no_o_id = @o_id

-- set carrier_id on this order (and get customer
id)

    update orders
    set o_carrier_id =
@o_carrier_id,
        @c_id = o_c_id
    where o_w_id = @w_id and
          o_d_id = @d_id and
          o_id = @o_id

-- set date in all lineitems for this order (and sum
amounts)

    update order_line
    set ol_delivery_d =
getdate(),
        @total = @total +
ol_amount
    where ol_w_id = @w_id and
          ol_d_id = @d_id and
          ol_o_id = @o_id

-- accumulate lineitem amounts for this order into
customer

    update customer
    set c_balance =
c_balance + @total,
        c_delivery_cnt =
c_delivery_cnt + 1
    where c_w_id = @w_id and
          c_d_id = @d_id and
          c_id = @c_id

  end

```

```

    select @oid1 = case @d_id when 1 then @o_id
else @oid1 end,
           @oid2 = case @d_id when 2 then @o_id
else @oid2 end,
           @oid3 = case @d_id when 3 then @o_id
else @oid3 end,
           @oid4 = case @d_id when 4 then @o_id
else @oid4 end,
           @oid5 = case @d_id when 5 then @o_id
else @oid5 end,
           @oid6 = case @d_id when 6 then @o_id
else @oid6 end,
           @oid7 = case @d_id when 7 then @o_id
else @oid7 end,
           @oid8 = case @d_id when 8 then @o_id
else @oid8 end,
           @oid9 = case @d_id when 9 then @o_id
else @oid9 end,
           @oid10 = case @d_id when 10 then @o_id
else @oid10 end

  end

commit tran d

-- return delivery data to client

select @oid1,
       @oid2,
       @oid3,
       @oid4,
       @oid5,
       @oid6,
       @oid7,
       @oid8,
       @oid9,
       @oid10

go

```

idxcuscl.sql

```

-- File:      IDXCUSCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on customer
table

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'customer_cl' )
drop index customer.customer_cl

```

```

create unique clustered index customer_cl on
customer(c_w_id, c_d_id, c_id)
on big_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

go

```

idxcusnc.sql

```

-- File:      IDXCUSNC.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates non-clustered index on customer
table

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'customer_nc1' )
drop index customer.customer_nc1

create unique nonclustered index customer_nc1 on
customer(c_w_id, c_d_id, c_last, c_first, c_id)
on big_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

go

```

idxdiscl.sql

```

-- File:      IDXDISCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on district
table

use tpcc
go

declare @startdate datetime
declare @enddate datetime

```

```

select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'district_cl' )
    drop index district.district_cl

create unique clustered index district_cl on
district(d_w_id, d_id)
    with fillfactor=100 on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

go

```

idxitmcl.sql

```

-- File:      IDXITMCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on item table

```

```

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'item_cl' )
    drop index item.item_cl

create unique clustered index item_cl on item(i_id)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

go

```

idxnodcl.sql

```

-- File:      IDXNODCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on new_order
table

```

```

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'new_order_cl' )
    drop index new_order.new_order_cl

create unique clustered index new_order_cl on
new_order(no_w_id, no_d_id, no_o_id)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

go

```

idxodlcl.sql

```

-- File:      IDXODLCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on order_line
table

```

```

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'order_line_cl' )
    drop index order_line.order_line_cl

create unique clustered index order_line_cl on
order_line(ol_w_id, ol_d_id, ol_o_id, ol_number)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

go

```

idxordcl.sql

```

-- File:      IDXORDCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on orders table

```

```

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'orders_cl' )
    drop index orders.orders_cl

create unique clustered index orders_cl on
orders(o_w_id, o_d_id, o_id)
    on misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

go

```

idxordnc.sql

```

-- File:      IDXORDNC.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates non-clustered index on orders
table

```

```

use tpcc
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

if exists ( select name from sysindexes where name =
'orders_nc1' )
    drop index orders.orders_nc1

create index orders_nc1 on orders(o_w_id, o_d_id,
o_c_id, o_id)
    on misc_fg

```

```

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

```

go

idxstkcl.sql

```

-- File:      IDXSTKCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on stock table

```

```

use tpcc
go

```

```

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

```

```

if exists ( select name from sysindexes where name =
'stock_cl' )
    drop index stock.stock_cl

```

```

create unique clustered index stock_cl on
stock(s_i_id, s_w_id)
    on big_fg

```

```

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

```

go

idxwarcl.sql

```

-- File:      IDXWARCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on warehouse
table

```

```

use tpcc
go

```

```

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert(varchar(30),@startdate,9)

```

```

if exists ( select name from sysindexes where name =
'warehouse_cl' )

```

```

drop index warehouse.warehouse_cl

```

```

create unique clustered index warehouse_cl on
warehouse(w_id)
    with fillfactor=100 on misc_fg

```

```

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)

```

go

neword.sql

```

-- File:      NEWORD.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates new order transaction stored
procedure
--
-- Modified 6/24/98 - Jamie Reding - Microsoft
Corporation
--           Replaced Select of @s_quantity with Select
of @li_qty to insure
--           correct data when sending line-item data to
client.
--

```

```

use tpcc
go

```

```

if exists ( select name from sysobjects where name =
"tpcc_neworder" )
    drop procedure tpcc_neworder
go

```

```

create proc tpcc_neworder

```

```

        @w_id          smallint,
        @d_id          tinyint,
        @c_id          int,
        @o_ol_cnt      tinyint,
        @o_all_local   tinyint,
        @i_id1 int = 0, @s_w_id1
smallint = 0, @ol_qty1 smallint = 0,
        @i_id2 int = 0, @s_w_id2
smallint = 0, @ol_qty2 smallint = 0,
        @i_id3 int = 0, @s_w_id3
smallint = 0, @ol_qty3 smallint = 0,
        @i_id4 int = 0, @s_w_id4
smallint = 0, @ol_qty4 smallint = 0,

```

```

        @i_id5 int = 0, @s_w_id5
smallint = 0, @ol_qty5 smallint = 0,
        @i_id6 int = 0, @s_w_id6
smallint = 0, @ol_qty6 smallint = 0,
        @i_id7 int = 0, @s_w_id7
smallint = 0, @ol_qty7 smallint = 0,
        @i_id8 int = 0, @s_w_id8
smallint = 0, @ol_qty8 smallint = 0,
        @i_id9 int = 0, @s_w_id9
smallint = 0, @ol_qty9 smallint = 0,
        @i_id10 int = 0, @s_w_id10
smallint = 0, @ol_qty10 smallint = 0,
        @i_id11 int = 0, @s_w_id11
smallint = 0, @ol_qty11 smallint = 0,
        @i_id12 int = 0, @s_w_id12
smallint = 0, @ol_qty12 smallint = 0,
        @i_id13 int = 0, @s_w_id13
smallint = 0, @ol_qty13 smallint = 0,
        @i_id14 int = 0, @s_w_id14
smallint = 0, @ol_qty14 smallint = 0,
        @i_id15 int = 0, @s_w_id15
smallint = 0, @ol_qty15 smallint = 0

```

```

as
declare @w_tax          numeric(4,4),
        @d_tax          numeric(4,4),
        @c_last        char(16),
        @c_credit       char(2),
        @c_discount     numeric(4,4),
        @i_price        numeric(5,2),
        @i_name         char(24),
        @i_data         char(50),
        @o_entry_d      datetime,
        @remote_flag    int,
        @s_quantity     smallint,
        @s_data         char(50),
        @s_dist         char(24),
        @li_no          int,
        @o_id           int,
        @commit_flag    tinyint,
        @li_id         int,
        @li_s_w_id     smallint,
        @li_qty        smallint,
        @ol_number      int,
        @c_id_local    int
begin

```

```

begin transaction n
-- get district tax and next available order id and
update
-- plus initialize local variables
update district
set @d_tax      = d_tax,
  @o_id        = d_next_o_id,
  d_next_o_id = d_next_o_id + 1,
  @o_entry_d   = getdate(),
  @li_no       = 0,
  @commit_flag = 1
where d_w_id = @w_id and
      d_id   = @d_id
-- process orderlines
while (@li_no < @o_ol_cnt)
begin
select @li_no = @li_no + 1
-- set i_id, s_w_id, and qty for this lineitem
select @li_id = case @li_no
  when 1 then @i_id1
  when 2 then @i_id2
  when 3 then @i_id3
  when 4 then @i_id4
  when 5 then @i_id5
  when 6 then @i_id6
  when 7 then @i_id7
  when 8 then @i_id8
  when 9 then @i_id9
  when 10 then
@i_id10
  when 11 then
@i_id11
  when 12 then
@i_id12
  when 13 then
@i_id13
  when 14 then
@i_id14
  when 15 then
@i_id15
end,
@li_s_w_id = case
  when 1 then
@s_w_id1
  when 2 then
@s_w_id2
  when 3 then
@s_w_id3
  when 4 then
@s_w_id4
  when 5 then
@s_w_id5
  when 6 then
@s_w_id6

```

```

@s_w_id7
@s_w_id8
@s_w_id9
@s_w_id10
@s_w_id11
@s_w_id12
@s_w_id13
@s_w_id14
@s_w_id15
@li_no
@ol_qty1
@ol_qty2
@ol_qty3
@ol_qty4
@ol_qty5
@ol_qty6
@ol_qty7
@ol_qty8
@ol_qty9
@ol_qty10
@ol_qty11
@ol_qty12
@ol_qty13
@ol_qty14
@ol_qty15
-- get item data (no one updates item)
select @i_price = i_price,
  @i_name      = i_name,
  @i_data      = i_data
  from item (tablock
repeatableread)
  where i_id = @li_id

```

```

when 7 then
when 8 then
when 9 then
when 10 then
when 11 then
when 12 then
when 13 then
when 14 then
when 15 then
end,
@li_qty = case
  when 1 then
  when 2 then
  when 3 then
  when 4 then
  when 5 then
  when 6 then
  when 7 then
  when 8 then
  when 9 then
  when 10 then
  when 11 then
  when 12 then
  when 13 then
  when 14 then
  when 15 then
end

```

```

-- if there actually is an item with this id, go to
work
      if (@@rowcount > 0)
      begin
update stock set s_ytd      = s_ytd +
@li_qty,
@s_quantity =
s_quantity,
@s_quantity =
s_quantity - @li_qty +
(s_quantity - @li_qty < 10) then 91 else 0 end,
s_order_cnt =
s_order_cnt + 1,
s_remote_cnt =
s_remote_cnt + case
  when (@li_s_w_id =
@w_id) then 0 else 1 end,
@s_data      = s_data,
@s_dist      = case
  @d_id
  when 1 then s_dist_01
  when 2 then s_dist_02
  when 3 then s_dist_03
  when 4 then s_dist_04
  when 5 then s_dist_05
  when 6 then s_dist_06
  when 7 then s_dist_07
  when 8 then s_dist_08
  when 9 then s_dist_09
  when 10 then s_dist_10
end
  where s_i_id = @li_id and
        s_w_id =
@li_s_w_id
-- insert order_line data (using data from item and
stock)
insert into order_line values(@o_id,
@d_id,
@s_w_id,
@li_no,
@li_id,
@li_s_w_id,
"dec 31",
@li_qty,
@i_price *
@li_qty,
@s_dist)

```

```

-- send line-item data to client
        select @i_name,
               @li_qty,
               b_g = case when (
(patindex("%ORIGINAL%",@i_data) > 0) and
(patindex("%ORIGINAL%",@s_data) > 0) )
               then "B" else "G" end,
               @i_price,
               @i_price * @li_qty
        end
        else
                begin
-- no item found - triggers rollback condition
                select "",0,"",0,0
                select @commit_flag = 0
        end
        end

-- get customer last name, discount, and credit
rating
        select @c_last      = c_last,
               @c_discount = c_discount,
               @c_credit    = c_credit,
               @c_id_local  = c_id
        from customer (repeatableread)
        where c_id = @c_id and
               c_w_id = @w_id and
               c_d_id = @d_id

-- insert fresh row into orders table
        insert into orders values (@o_id,
                                   @d_id,
                                   @w_id,
                                   @c_id_local,
                                   @o_entry_d,
                                   0,
                                   @o_ol_cnt,
                                   @o_all_local)

-- insert corresponding row into new-order table
        insert into new_order values (@o_id,
                                       @d_id,
                                       @w_id)

```

```

-- select warehouse tax
        select @w_tax = w_tax
        from warehouse (repeatableread)
        where w_id = @w_id
        if (@commit_flag = 1)
                commit transaction n
        else
-- all that work for nuthin!!!
                rollback transaction n

-- return order data to client
        select @w_tax,
               @d_tax,
               @o_id,
               @c_last,
               @c_discount,
               @c_credit,
               @o_entry_d,
               @commit_flag
        end
        go



---


ordstat.sql


---


-- File:      ORDSTAT.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates order status transaction stored
procedure

use tpcc
go

if exists ( select name from sysobjects where name =
"tpcc_orderstatus" )
        drop procedure    tpcc_orderstatus
go

create proc tpcc_orderstatus @w_id
        smallint,
               @d_id                tinyint,
               @c_id                int,
               @c_last char(16) = ""
        as

declare @c_balance        numeric(12,2),

```

```

        @c_first          char(16),
        @c_middle         char(2),
        @o_id             int,
        @o_entry_d        datetime,
        @o_carrier_id     smallint,
        @cnt              smallint

begin tran o
        if (@c_id = 0)
                begin
-- get customer id and info using last name
                select @cnt = (count(*)+1)/2
                from customer (repeatableread)
                where c_last = @c_last and
                       c_w_id = @w_id and
                       c_d_id = @d_id
                set rowcount @cnt
                select @c_id = c_id,
                       @c_balance =
        c_balance,
                       @c_first =
        c_first,
                       @c_last = c_last,
                       @c_middle =
        c_middle
                from customer (repeatableread)
                where c_last = @c_last and
                       c_w_id = @w_id and
                       c_d_id = @d_id
                order by c_w_id, c_d_id, c_last,
        c_first
                set rowcount 0
                end
        else
                begin
-- get customer info if by id
                select @c_balance = c_balance,
                       @c_first = c_first,
                       @c_middle = c_middle,
                       @c_last = c_last
                from customer (repeatableread)
                where c_id = @c_id and
                       c_d_id = @d_id and
                       c_w_id = @w_id
                select @cnt = @@rowcount
                end
-- if no such customer
                if (@cnt = 0)
                        begin

```

```

raiserror("Customer not
found",18,1)
        goto custnotfound
    end
-- get order info
        select @o_id = o_id,
               @o_entry_d = o_entry_d,
               @o_carrier_id = o_carrier_id
        from orders (serializable)
        where o_c_id = @c_id and
              o_d_id = @d_id and
              o_w_id = @w_id
        order by o_id asc
-- select order lines for the current order
        select ol_supply_w_id,
               ol_i_id,
               ol_quantity,
               ol_amount,
               ol_delivery_d
        from order_line (repeatable)
        where ol_o_id = @o_id and
              ol_d_id = @d_id and
              ol_w_id = @w_id
custnotfound:
commit tran o
-- return data to client
select @c_id,
       @c_last,
       @c_first,
       @c_middle,
       @o_entry_d,
       @o_carrier_id,
       @c_balance,
       @o_id
go

```

payment.sql

```

-- File:      PAYMENT.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.00
-- Copyright Microsoft, 1996
-- Purpose:   Creates payment transaction stored
--           procedure
use tpcc
go
if exists (select name from sysobjects where name =
"tpcc_payment" )
    drop procedure tpcc_payment

```

```

go
create proc tpcc_payment @w_id          smallint,
                        @c_w_id        smallint,
                        @h_amount       numeric(6,2),
                        @d_id           tinyint,
                        @c_d_id         tinyint,
                        @c_id           int,
                        @c_last         char(16) = ""
as
declare @w_street_1    char(20),
        @w_street_2    char(20),
        @w_city        char(20),
        @w_state       char(2),
        @w_zip         char(9),
        @w_name        char(10),
        @d_street_1    char(20),
        @d_street_2    char(20),
        @d_city        char(20),
        @d_state       char(2),
        @d_zip         char(9),
        @d_name        char(10),
        @c_first       char(16),
        @c_middle      char(2),
        @c_street_1    char(20),
        @c_street_2    char(20),
        @c_city        char(20),
        @c_state       char(2),
        @c_zip         char(9),
        @c_phone       char(16),
        @c_since       datetime,
        @c_credit       char(2),
        @c_credit_lim   numeric(12,2),
        @c_balance     numeric(12,2),
        @c_discount    numeric(4,4),
        @data          char(500),
        @c_data        char(500),
        @datetime      datetime,
        @w_ytd         numeric(12,2),
        @d_ytd         numeric(12,2),
        @cnt           smallint,
        @val           smallint,
        @screen_data   char(200),
        @d_id_local   tinyint,
        @w_id_local   smallint,
        @c_id_local   int
select @screen_data = ""
begin tran p
-- get payment date

```

```

select @datetime = getdate()
if (@c_id = 0)
begin
-- get customer id and info using last name
        select @cnt = count(*)
        from customer (repeatable)
        where c_last = @c_last and
              c_w_id = @c_w_id and
              c_d_id = @c_d_id
        select @val = (@cnt + 1) / 2
        set rowcount @val
        select @c_id = c_id
        from customer (repeatable)
        where c_last = @c_last and
              c_w_id = @c_w_id and
              c_d_id = @c_d_id
        order by c_last, c_first
        set rowcount 0
    end
-- get customer info and update balances
        update customer set
           @c_balance = c_balance =
c_balance - @h_amount,
           c_payment_cnt = c_payment_cnt +
1,
           c_ytd_payment = c_ytd_payment +
@h_amount,
           @c_first = c_first,
           @c_middle = c_middle,
           @c_last = c_last,
           @c_street_1 = c_street_1,
           @c_street_2 = c_street_2,
           @c_city = c_city,
           @c_state = c_state,
           @c_zip = c_zip,
           @c_phone = c_phone,
           @c_credit = c_credit,
           @c_credit_lim = c_credit_lim,
           @c_discount = c_discount,
           @c_since = c_since,
           @data = c_data,
           @c_id_local = c_id
        where c_id = @c_id and
              c_w_id = @c_w_id and
              c_d_id = @c_d_id
-- if customer has bad credit get some more info
        if (@c_credit = "BC")
        begin
-- compute new info

```



```

        select @c_data =
convert(char(5),@c_id) +
convert(char(4),@c_d_id) +
convert(char(5),@c_w_id) +
convert(char(4),@d_id) +
convert(char(5),@w_id) +
convert(char(19),@h_amount) +
substring(@data, 1, 458)
-- update customer info
        update customer set
            c_data = @c_data
        where c_id = @c_id and
            c_w_id = @c_w_id and
            c_d_id = @c_d_id
        select @screen_data = substring
(@c_data,1,200)
        end
-- get district data and update year-to-date
        update district
set d_ytd = d_ytd + @h_amount,
@d_street_1 = d_street_1,
@d_street_2 = d_street_2,
@d_city = d_city,
@d_state = d_state,
@d_zip = d_zip,
@d_name = d_name,
@d_id_local = d_id
where d_w_id = @w_id and
d_id = @d_id
-- get warehouse data and update year-to-date
        update warehouse
set w_ytd = w_ytd + @h_amount,
w_street_1 = w_street_1,
w_street_2 = w_street_2,
w_city = w_city,
w_state = w_state,
w_zip = w_zip,
w_name = w_name,
w_id_local = w_id
where w_id = @w_id
-- create history record
        insert into history values (@c_id_local,
            @c_d_id,
            @c_w_id,

```

```

            @d_id_local,
            @w_id_local,
            @datetime,
            @h_amount,
            @w_name + " " +
@d_name)
        commit tran p
-- return data to client
select @c_id,
        @c_last,
        @datetime,
        @w_street_1,
        @w_street_2,
        @w_city,
        @w_state,
        @w_zip,
        @d_street_1,
        @d_street_2,
        @d_city,
        @d_state,
        @d_zip,
        @c_first,
        @c_middle,
        @c_street_1,
        @c_street_2,
        @c_city,
        @c_state,
        @c_zip,
        @c_phone,
        @c_since,
        @c_credit,
        @c_credit_lim,
        @c_discount,
        @c_balance,
        @screen_data
go

```

stocklev.sql

```

-- File: STOCKLEV.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.00
-- Copyright Microsoft, 1996
-- Purpose: Creates stock level transaction stored
procedure
use tpcc
go
if exists (select name from sysobjects where name =
"tpcc_stocklevel" )
        drop procedure tpcc_stocklevel
go

```

```

create proc tpcc_stocklevel @w_id
        smallint,
        @d_id tinyint,
        @threshold smallint
as
        declare @o_id_low int,
            @o_id_high int
        select @o_id_low = (d_next_o_id - 20),
            @o_id_high = (d_next_o_id - 1)
        from district
        where d_w_id = @w_id and
            d_id = @d_id
        select count(distinct(s_i_id))
        from stock, order_line
        where ol_w_id = @w_id and
            ol_d_id = @d_id and
            ol_o_id between @o_id_low and
@o_id_high and
            s_w_id = ol_w_id and
            s_i_id = ol_i_id and
            s_quantity < @threshold
go

```

tables.sql

```

-- File: TABLES.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.00
-- Copyright Microsoft, 1996
-- Purpose: Creates TPC-C tables
use tpcc
go
if exists ( select name from sysobjects where name =
'warehouse' )
        drop table warehouse
go
create table warehouse
(
        w_id
        smallint,
        w_name
        char(10),
        w_street_1
        char(20),
        w_street_2
        char(20),
        w_city
        char(20),
        w_state
        char(2),
        w_zip
        char(9),

```

```

        w_tax
        numeric(4,4),
        w_ytd
        numeric(12,2)
    ) on misc_fg
go

if exists ( select name from sysobjects where name =
'district' )
    drop table district
go
create table district
(
    d_id
    tinyint,
    d_w_id
    smallint,
    d_name
    char(10),
    d_street_1
    char(20),
    d_street_2
    char(20),
    d_city
    char(20),
    d_state
    char(2),
    d_zip
    char(9),
    d_tax
    numeric(4,4),
    d_ytd
    numeric(12,2),
    d_next_o_id
    int
) on misc_fg
go

if exists ( select name from sysobjects where name =
'customer' )
    drop table customer
go
create table customer
(
    c_id
    int,
    c_d_id
    tinyint,
    c_w_id
    smallint,
    c_first
    char(16),
    c_middle
    char(2),
    c_last
    char(16),
    c_street_1
    char(20),
    c_street_2
    char(20),
    c_city
    char(20),
    c_state
    char(2),

```

```

        c_zip
        char(9),
        c_phone
        char(16),
        c_since
        datetime,
        c_credit
        char(2),
        c_credit_lim
        numeric(12,2),
        c_discount
        numeric(4,4),
        c_balance
        numeric(12,2),
        c_ytd_payment
        numeric(12,2),
        c_payment_cnt
        smallint,
        c_delivery_cnt
        smallint,
        c_data
        char(500)
    ) on big_fg
go

if exists ( select name from sysobjects where name =
'history' )
    drop table history
go
create table history
(
    h_c_id
    int,
    h_c_d_id
    tinyint,
    h_c_w_id
    smallint,
    h_d_id
    tinyint,
    h_w_id
    smallint,
    h_date
    datetime,
    h_amount
    numeric(6,2),
    h_data
    char(24)
) on misc_fg
go

if exists ( select name from sysobjects where name =
'new_order' )
    drop table new_order
go
create table new_order
(
    no_o_id
    int,
    no_d_id
    tinyint,
    no_w_id
    smallint
) on misc_fg
go

if exists ( select name from sysobjects where name =
'orders' )
    drop table orders
go

```

```

create table orders
(
    o_id
    int,
    o_d_id
    tinyint,
    o_w_id
    smallint,
    o_c_id
    int,
    o_entry_d
    datetime,
    o_carrier_id
    tinyint,
    o_ol_cnt
    tinyint,
    o_all_local
    tinyint
) on misc_fg
go

if exists ( select name from sysobjects where name =
'order_line' )
    drop table order_line
go
create table order_line
(
    ol_o_id
    int,
    ol_d_id
    tinyint,
    ol_w_id
    smallint,
    ol_number
    tinyint,
    ol_i_id
    int,
    ol_supply_w_id
    smallint,
    ol_delivery_d
    datetime,
    ol_quantity
    smallint,
    ol_amount
    numeric(6,2),
    ol_dist_info
    char(24)
) on misc_fg
go

if exists ( select name from sysobjects where name =
'item' )
    drop table item
go
create table item
(
    i_id
    int,
    i_im_id
    int,
    i_name
    char(24),
    i_price
    numeric(5,2),
    i_data
    char(50)
) on misc_fg
go

if exists ( select name from sysobjects where name =
'stock' )

```

```

drop table stock
go
create table stock
(
    s_i_id
    int,
    s_w_id
    smallint,
    s_quantity
    smallint,
    s_dist_01
    char(24),
    s_dist_02
    char(24),
    s_dist_03
    char(24),
    s_dist_04
    char(24),
    s_dist_05
    char(24),
    s_dist_06
    char(24),
    s_dist_07
    char(24),
    s_dist_08
    char(24),
    s_dist_09
    char(24),
    s_dist_10
    char(24),
    s_ytd
    int,
    s_order_cnt
    smallint,
    s_remote_cnt
    smallint,
    s_data
    char(50)
) on big_fg
go

```

createdb.sql

```

-- File:      CREATEDB.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates tpcc database and backup files

use master
go

-- remove any existing database and backup files

exec sp_dbremove tpcc, dropdev
exec sp_dropdevice 'tpccback1'
exec sp_dropdevice 'tpccback2'
exec sp_dropdevice 'tpccback3'
exec sp_dropdevice 'tpccback4'
go

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:",
convert (varchar(30),@startdate,9)

-- create main database files

create database tpcc on

```

```

(name="MSSQL70_tpcc_root",filename="C:\MSSQ
L70_tpcc_root.mdf",size=8MB, FILEGROWTH=0)
log on
(name="MSSQL70_tpcc_log",filename="F:",size
=52000MB, FILEGROWTH=0)

-- create filegroups
alter database tpcc add filegroup misc_fg
alter database tpcc add filegroup big_fg

-- add files to filegroups
-- 10028 * 7
-- 8775 * 8
alter database tpcc add file
(name="misc1",filename="N:",size=12000MB,
FILEGROWTH=0),
(name="misc2",filename="O:",size=12000MB,
FILEGROWTH=0),
(name="misc3",filename="P:",size=12000MB,
FILEGROWTH=0),
(name="misc4",filename="Q:",size=12000MB,
FILEGROWTH=0),
(name="misc5",filename="R:",size=12000MB,
FILEGROWTH=0),
(name="misc6",filename="S:",size=12000MB,
FILEGROWTH=0),
(name="misc7",filename="T:",size=12000MB,
FILEGROWTH=0)
to filegroup misc_fg

-- 15425 * 7
alter database tpcc add file
(name="stock1",filename="G:",size=17000MB,
FILEGROWTH=0),
(name="stock2",filename="H:",size=17000MB,
FILEGROWTH=0),
(name="stock3",filename="I:",size=17000MB,
FILEGROWTH=0),
(name="stock4",filename="J:",size=17000MB,
FILEGROWTH=0),
(name="stock5",filename="K:",size=17000MB,
FILEGROWTH=0),
(name="stock6",filename="L:",size=17000MB,
FILEGROWTH=0),
(name="stock7",filename="M:",size=17000MB,
FILEGROWTH=0)
to filegroup big_fg

select @enddate = getdate()
select "End date: ", convert (varchar(30),@enddate,9)
select "Elapsed time (in seconds): ",
datediff(second, @startdate, @enddate)
go

-- create backup devices

```

```

exec sp_addumpdevice
'disk','tpccback1','y:\tpccback1.dmp'
exec sp_addumpdevice
'disk','tpccback2','y:\tpccback2.dmp'
exec sp_addumpdevice
'disk','tpccback3','z:\tpccback3.dmp'
exec sp_addumpdevice
'disk','tpccback4','z:\tpccback4.dmp'
go

```

tpccldr.mak

```

# Microsoft Developer Studio Generated NMAKE File,
Format Version 4.10
# ** DO NOT EDIT **

# TARGETTYPE "Win32 (x86) Console Application" 0x0103

!IF "$(CFG)" == ""
CFG=tpccldr - Win32 Debug
!MESSAGE No configuration specified. Defaulting to
tpccldr - Win32 Debug.
!ENDIF

!IF "$(CFG)" != "tpccldr - Win32 Release" && "$(CFG)"
!=\
"tpccldr - Win32 Debug"
!MESSAGE Invalid configuration "$(CFG)" specified.
!MESSAGE You can specify a configuration when running
NMAKE on this makefile
!MESSAGE by defining the macro CFG on the command
line. For example:
!MESSAGE
!MESSAGE NMAKE /f "tpccldr.mak" CFG="tpccldr - Win32
Debug"
!MESSAGE
!MESSAGE Possible choices for configuration are:
!MESSAGE
!MESSAGE "tpccldr - Win32 Release" (based on "Win32
(x86) Console Application")
!MESSAGE "tpccldr - Win32 Debug" (based on "Win32
(x86) Console Application")
!MESSAGE
!ERROR An invalid configuration is specified.
!ENDIF

!IF "$(OS)" == "Windows_NT"
NULL=
!ELSE
NULL=nul
!ENDIF
#####
# Begin Project
# PROP Target_Last_Scanned "tpccldr - Win32 Debug"
RSC=rc.exe
CPP=cl.exe

!IF "$(CFG)" == "tpccldr - Win32 Release"

# PROP BASE Use_MFC 0
# PROP BASE Use_Debug_Libraries 0

```

```

# PROP BASE Output_Dir "Release"
# PROP BASE Intermediate_Dir "Release"
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 0
# PROP Output_Dir "bin"
# PROP Intermediate_Dir "objects"
# PROP Target_Dir ""
OUTDIR=. \bin
INTDIR=. \objects

ALL : "$(OUTDIR)\tpccldr.exe"

CLEAN :
    -@erase "$(INTDIR)\getargs.obj"
    -@erase "$(INTDIR)\random.obj"
    -@erase "$(INTDIR)\strings.obj"
    -@erase "$(INTDIR)\time.obj"
    -@erase "$(INTDIR)\tpccldr.obj"
    -@erase "$(OUTDIR)\tpccldr.exe"

"$(OUTDIR)" :
    if not exist "$(OUTDIR)\$(NULL)" mkdir
"$(OUTDIR)"

"$(INTDIR)" :
    if not exist "$(INTDIR)\$(NULL)" mkdir
"$(INTDIR)"

# ADD BASE CPP /nologo /W3 /GX /O2 /D "WIN32" /D
"NDEBUG" /D "_CONSOLE" /YX /c
# ADD CPP /nologo /MT /W3 /GX /O2 /I
"c:\mssql\dblib\include" /D "NDEBUG" /D "WIN32" /D
"_CONSOLE" /D "DBNTWIN32" /c
# SUBTRACT CPP /YX
CPP_PROJ=/nologo /MT /W3 /GX /O2 /I
"c:\mssql\dblib\include" /D "NDEBUG" /D\
"WIN32" /D "_CONSOLE" /D "DBNTWIN32" /Fo"$(INTDIR)"/
/c
CPP_OBJS=. \objects/
CPP_SBRS=.\
# ADD BASE RSC /1 0x409 /d "NDEBUG"
# ADD RSC /1 0x409 /d "NDEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
BSC32_FLAGS=/nologo /o"$(OUTDIR)\tpccldr.bsc"
BSC32_SBRS= \

LINK32=link.exe
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib
winspool.lib comdlg32.lib advapi32.lib shell32.lib
ole32.lib oleaut32.lib uuid.lib odbcc32.lib
odbccp32.lib /nologo /subsystem:console /machine:I386
# ADD LINK32 c:\mssql\dblib\lib\ntwdblib.lib
kernel32.lib user32.lib gdi32.lib winspool.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib uuid.lib odbcc32.lib odbccp32.lib /nologo
/subsystem:console /pdb:none /debug /machine:I386
LINK32_FLAGS=c:\mssql\dblib\lib\ntwdblib.lib
kernel32.lib user32.lib gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib
ole32.lib oleaut32.lib\

```

```

uuid.lib odbcc32.lib odbccp32.lib /nologo
/subsystem:console /pdb:none\
/machine:I386 /out:"$(OUTDIR)\tpccldr.exe"
LINK32_OBJS= \
    "$(INTDIR)\getargs.obj" \
    "$(INTDIR)\random.obj" \
    "$(INTDIR)\strings.obj" \
    "$(INTDIR)\time.obj" \
    "$(INTDIR)\tpccldr.obj"

"$(OUTDIR)\tpccldr.exe" : "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
    $(LINK32) @<<
    $(LINK32_FLAGS) $(LINK32_OBJS)
<<

!ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"

# PROP BASE Use_MFC 0
# PROP BASE Use_Debug_Libraries 1
# PROP BASE Output_Dir "Debug"
# PROP BASE Intermediate_Dir "Debug"
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 1
# PROP Output_Dir "bin"
# PROP Intermediate_Dir "objects"
# PROP Target_Dir ""
OUTDIR=. \bin
INTDIR=. \objects

ALL : "$(OUTDIR)\tpccldr.exe"

CLEAN :
    -@erase "$(INTDIR)\getargs.obj"
    -@erase "$(INTDIR)\random.obj"
    -@erase "$(INTDIR)\strings.obj"
    -@erase "$(INTDIR)\time.obj"
    -@erase "$(INTDIR)\tpccldr.obj"
    -@erase "$(INTDIR)\vc40.idb"
    -@erase "$(INTDIR)\vc40.pdb"
    -@erase "$(OUTDIR)\tpccldr.exe"

"$(OUTDIR)" :
    if not exist "$(OUTDIR)\$(NULL)" mkdir
"$(OUTDIR)"

"$(INTDIR)" :
    if not exist "$(INTDIR)\$(NULL)" mkdir
"$(INTDIR)"

# ADD BASE CPP /nologo /W3 /Gm /GX /Zi /Od /D "WIN32"
/D "DEBUG" /D "_CONSOLE" /YX /c
# ADD CPP /nologo /MTd /W3 /Gm /GX /Zi /Od /I
"c:\mssql\dblib\include" /D "_DEBUG" /D "WIN32" /D
"_CONSOLE" /D "DBNTWIN32" /c
# SUBTRACT CPP /YX
CPP_PROJ=/nologo /MTd /W3 /Gm /GX /Zi /Od /I
"c:\mssql\dblib\include" /D\
"DEBUG" /D "WIN32" /D "_CONSOLE" /D "DBNTWIN32"
/Fo"$(INTDIR)"/\
/Fd"$(INTDIR)"/\ /c
CPP_OBJS=. \objects/

```

```

CPP_SBRS=.\
# ADD BASE RSC /1 0x409 /d "DEBUG"
# ADD RSC /1 0x409 /d "DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
BSC32_FLAGS=/nologo /o"$(OUTDIR)\tpccldr.bsc"
BSC32_SBRS= \

LINK32=link.exe
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib
winspool.lib comdlg32.lib advapi32.lib shell32.lib
ole32.lib oleaut32.lib uuid.lib odbcc32.lib
odbccp32.lib /nologo /subsystem:console /debug
/machine:I386
# ADD LINK32 c:\mssql\dblib\lib\ntwdblib.lib
kernel32.lib user32.lib gdi32.lib
comdlg32.lib advapi32.lib shell32.lib ole32.lib
oleaut32.lib uuid.lib odbcc32.lib odbccp32.lib /nologo
/subsystem:console /pdb:none /debug /machine:I386
LINK32_FLAGS=c:\mssql\dblib\lib\ntwdblib.lib
kernel32.lib user32.lib gdi32.lib\
winspool.lib comdlg32.lib advapi32.lib shell32.lib
ole32.lib oleaut32.lib\
uuid.lib odbcc32.lib odbccp32.lib /nologo
/subsystem:console /pdb:none /debug\
/machine:I386 /out:"$(OUTDIR)\tpccldr.exe"
LINK32_OBJS= \
    "$(INTDIR)\getargs.obj" \
    "$(INTDIR)\random.obj" \
    "$(INTDIR)\strings.obj" \
    "$(INTDIR)\time.obj" \
    "$(INTDIR)\tpccldr.obj"

"$(OUTDIR)\tpccldr.exe" : "$(OUTDIR)" $(DEF_FILE)
$(LINK32_OBJS)
    $(LINK32) @<<
    $(LINK32_FLAGS) $(LINK32_OBJS)
<<

!ENDIF

.c{$(CPP_OBJS)}.obj:
    $(CPP) $(CPP_PROJ) $<

.cpp{$(CPP_OBJS)}.obj:
    $(CPP) $(CPP_PROJ) $<

.cxx{$(CPP_OBJS)}.obj:
    $(CPP) $(CPP_PROJ) $<

.c{$(CPP_SBRS)}.sbr:
    $(CPP) $(CPP_PROJ) $<

.cpp{$(CPP_SBRS)}.sbr:
    $(CPP) $(CPP_PROJ) $<

.cxx{$(CPP_SBRS)}.sbr:
    $(CPP) $(CPP_PROJ) $<

#####
# Begin Target

```

```

# Name "tpccldr - Win32 Release"
# Name "tpccldr - Win32 Debug"

!IF "$(CFG)" == "tpccldr - Win32 Release"
!ELSEIF "$(CFG)" == "tpccldr - Win32 Debug"
!ENDIF

#####
# Begin Source File

SOURCE=.\src\random.c
DEP_CPP_RAND0=\
    ".\src\tpcc.h"\
    "\mssql\dblib\include\sqldb.h"\
    "\mssql\dblib\include\sqlfront.h\"

"$ (INTDIR)\random.obj" : $(SOURCE) $(DEP_CPP_RAND0)
"$ (INTDIR)"
    $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
# Begin Source File

SOURCE=.\src\strings.c
DEP_CPP_STRIN=\
    ".\src\tpcc.h"\
    "\mssql\dblib\include\sqldb.h"\
    "\mssql\dblib\include\sqlfront.h\"

"$ (INTDIR)\strings.obj" : $(SOURCE) $(DEP_CPP_STRIN)
"$ (INTDIR)"
    $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
# Begin Source File

SOURCE=.\src\time.c
DEP_CPP_TIME_=\
    ".\src\tpcc.h"\
    "\mssql\dblib\include\sqldb.h"\
    "\mssql\dblib\include\sqlfront.h\"

"$ (INTDIR)\time.obj" : $(SOURCE) $(DEP_CPP_TIME_)
"$ (INTDIR)"
    $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####

```

```

# Begin Source File

SOURCE=.\src\tpccldr.c
DEP_CPP_TPCC=\
    ".\src\tpcc.h"\
    "\mssql\dblib\include\sqldb.h"\
    "\mssql\dblib\include\sqlfront.h\"

"$ (INTDIR)\tpccldr.obj" : $(SOURCE) $(DEP_CPP_TPCC)
"$ (INTDIR)"
    $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
#####
# Begin Source File

SOURCE=.\src\getargs.c
DEP_CPP_GETAR=\
    ".\src\tpcc.h"\
    "\mssql\dblib\include\sqldb.h"\
    "\mssql\dblib\include\sqlfront.h\"

"$ (INTDIR)\getargs.obj" : $(SOURCE) $(DEP_CPP_GETAR)
"$ (INTDIR)"
    $(CPP) $(CPP_PROJ) $(SOURCE)

# End Source File
# End Target
# End Project
#####

```

tpcc.h

```

// File: TPCC.H Microsoft
// TPC-C Kit Ver. 4.00 Copyright
// Microsoft, 1996, 1997, 1998 Purpose: Header file for TPC-C database loader

// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.00"

// General headers
#include <windows.h>
#include <winbase.h>
#include <stdlib.h>
#include <stdio.h>
#include <process.h>
#include <stddef.h>
#include <stdarg.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>

```

```

#include <sys\types.h>

// ODBC headers
#include <sql.h>
#include <sqlext.h>
#include <odbcss.h>

// General constants
#define MILLI 1000
#define FALSE 0
#define TRUE 1
#define UNDEF -1
#define MINPRINTASCII 32
#define MAXPRINTASCII 126

// Default environment constants
#define SERVER ""
#define DATABASE "tpcc"
#define USER "sa"
#define PASSWORD ""

// Default loader arguments
#define BATCH 10000
#define DEFLOADPACKSIZE 32768
#define ORDERS_PER_DIST 3000
#define LOADER_RES_FILE "logs\load.out"
#define LOADER_NURAND_C 123
#define DEF_STARTING_WAREHOUSE 1
#define BUILD_INDEX 1 // build both data and indexes
#define INDEX_ORDER 1 // build indexes before load
#define SCALE_DOWN 0 // build a normal scale database
#define INDEX_SCRIPT_PATH "scripts"

typedef struct
{
    char *server;
    char *database;
    char *user;
    char *password;
    BOOL tables_all; // set if loading all tables
    BOOL table_item; // set if loading ITEM table specifically
    BOOL table_warehouse; // set if loading WAREHOUSE, DISTRICT, and STOCK
}

```

```

        BOOL
        table_customer;           // set if
loading CUSTOMER and HISTORY
        BOOL
        table_orders;           // set if
loading NEW-ORDER, ORDERS, ORDER-LINE
        long
        num_warehouses;
        long
        batch;
        long
        verbose;
        long
        pack_size;
        char
        *loader_res_file;
        char
        *synch_servername;
        long
        case_sensitivity;
        long
        starting_warehouse;
        long
        build_index;
        long
        index_order;
        long
        scale_down;
        char
        *index_script_path;
} TPCCLDR_ARGS;

// String length constants
#define SERVER_NAME_LEN      20
#define DATABASE_NAME_LEN   20
#define USER_NAME_LEN       20
#define PASSWORD_LEN        20
#define TABLE_NAME_LEN     20
#define I_DATA_LEN          50
#define I_NAME_LEN           24
#define BRAND_LEN            1
#define LAST_NAME_LEN        16
#define W_NAME_LEN           10
#define ADDRESS_LEN          20
#define STATE_LEN            2
#define ZIP_LEN               9
#define S_DIST_LEN           24
#define S_DATA_LEN           50
#define D_NAME_LEN           10
#define FIRST_NAME_LEN       16
#define MIDDLE_NAME_LEN      2
#define PHONE_LEN            16
#define CREDIT_LEN           2
#define C_DATA_LEN           500
#define H_DATA_LEN           24
#define DIST_INFO_LEN        24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN           25
#define OL_DIST_INFO_LEN

24
        #define C_SINCE_LEN
        23

```

```

        #define H_DATE_LEN
        23
        #define OL_DELIVERY_D_LEN
23
        #define O_ENTRY_D_LEN           23

// Functions in random.c
void seed();
long irand();
double drand();
void WUCreate();
short WURand();
long RandomNumber(long lower, long upper);

// Functions in getargs.c;
void GetArgsLoader();
void GetArgsLoaderUsage();

// Functions in time.c
long TimeNow();

// Functions in strings.c
void MakeAddress();
void LastName();
int MakeAlphaString();
int MakeOriginalAlphaString();
int MakeNumberString();
int MakeZipNumberString();
void InitString();
void InitAddress();
void PaddString();

tpccldr.c
// File: TPCCLDR.C
// Microsoft
TPC-C Kit Ver. 4.00
// Copyright
Microsoft, 1996, 1997, 1998
// Purpose: Source file for TPC-C database
loader

// Includes
#include "tpcc.h"
#include "search.h"

// Defines
#define MAXITEMS 100000
#define MAXITEMS_SCALE_DOWN 100
#define CUSTOMERS_PER_DISTRICT 3000
#define CUSTOMERS_SCALE_DOWN 30
#define DISTRICT_PER_WAREHOUSE 10
#define ORDERS_PER_DISTRICT 3000
#define ORDERS_SCALE_DOWN 30
#define MAX_CUSTOMER_THREADS 2
#define MAX_ORDER_THREADS 3
#define MAX_MAIN_THREADS 4

// Functions declarations

```

```

void HandleErrorDBC (SQLHDBC hdbc1);

long NURand();
void LoadItem();
void LoadWarehouse();

void Stock();
void District();

void LoadCustomer();
void CustomerBufInit();
void CustomerBufLoad();
void LoadCustomerTable();
void LoadHistoryTable();

void LoadOrders();
void OrdersBufInit();
void OrdersBufLoad();
void LoadOrdersTable();
void LoadNewOrderTable();
void LoadOrderLineTable();
void GetPermutation();
void CheckForCommit();
void OpenConnections();
void BuildIndex();
void FormatDate ();

// Shared memory structures

typedef struct
{
        long ol;
        long ol_i_id;
        short ol_supply_w_id;
        short ol_quantity;
        double ol_amount;
        char
        ol_dist_info[DIST_INFO_LEN+1];
        char
        ol_delivery_d[OL_DELIVERY_D_LEN+1];
} ORDER_LINE_STRUCT;

typedef struct
{
        long o_id;
        short o_d_id;
        short o_w_id;
        long o_c_id;
        short o_carrier_id;
        short o_ol_cnt;
        short o_all_local;
        ORDER_LINE_STRUCT o_ol[15];
} ORDERS_STRUCT;

typedef struct
{
        long c_id;
        short c_d_id;
        short c_w_id;
        char
        c_first[FIRST_NAME_LEN+1];
        char
        c_middle[MIDDLE_NAME_LEN+1];

```

```

char
  c_last[LAST_NAME_LEN+1];
char
  c_street_1[ADDRESS_LEN+1];
char
  c_street_2[ADDRESS_LEN+1];
char
  c_city[ADDRESS_LEN+1];
char
  c_state[STATE_LEN+1];
char
  c_zip[ZIP_LEN+1];
char
  c_phone[PHONE_LEN+1];
char
  c_credit[CREDIT_LEN+1];
double
  c_credit_lim;
double
  c_discount;
// fix to avoid ODBC float to numeric conversion
// problem.
// double
  c_balance;
  char
  c_balance[6];

double
  c_ytd_payment;
short
  c_payment_cnt;
short
  c_delivery_cnt;
char
  c_data[C_DATA_LEN+1];
double
  h_amount;
char
  h_data[H_DATA_LEN+1];
} CUSTOMER_STRUCT;

typedef struct
{
  char
  c_last[LAST_NAME_LEN+1];
  char
  c_first[FIRST_NAME_LEN+1];
  long
  c_id;
} CUSTOMER_SORT_STRUCT;

typedef struct
{
  long
  time_start;
} LOADER_TIME_STRUCT;

// Global variables
char
  szLastError[300];
HENV
  henv;

```

```

HDBC
  i_hdbc1;
// for ITEM table
HDBC
  w_hdbc1;
// for WAREHOUSE, DISTRICT, STOCK
HDBC
  c_hdbc1;
// for CUSTOMER
HDBC
  c_hdbc2;
// for HISTORY
HDBC
  o_hdbc1;
// for ORDERS
HDBC
  o_hdbc2;
// for NEW-ORDER
HDBC
  o_hdbc3;
// for ORDER-LINE

HSTMT
  i_hstmt1;
HSTMT
  w_hstmt1;
HSTMT
  c_hstmt1, c_hstmt2;
HSTMT
  o_hstmt1, o_hstmt2, o_hstmt3;

ORDERS_STRUCT  orders_buf[ORDERS_PER_DISTRICT];
CUSTOMER_STRUCT customer_buf[CUSTOMERS_PER_DISTRICT];
long
  orders_rows_loaded;
long
  new_order_rows_loaded;
long
  order_line_rows_loaded;
long
  history_rows_loaded;
long
  customer_rows_loaded;
long
  stock_rows_loaded;
long
  district_rows_loaded;
long
  item_rows_loaded;
long
  warehouse_rows_loaded;
long
  main_time_start;
long
  main_time_end;
long
  max_items;
long
  customers_per_district;
long
  orders_per_district;
long
  first_new_order;
long
  last_new_order;

TPCCCLDR_ARGS  *aptr, args;

//=====
//
// Function name: main
//
//=====

int main(int argc, char **argv)
{
  DWORD
  dwThreadId[MAX_MAIN_THREADS];
  HANDLE
  hThread[MAX_MAIN_THREADS];
  FILE
  *fLoader;
  char
  buffer[255];
  int
  i;

  for (i=0; i<MAX_MAIN_THREADS; i++)

```

```

  hThread[i] = NULL;

printf("\n*****\n");
printf("\n*
*");
printf("\n* Microsoft SQL Server
*");
printf("\n*
*");
printf("\n* TPC-C BENCHMARK KIT: Database
loader *");
printf("\n* Version %s
*", TPCKIT_VER);
printf("\n*
*");
printf("\n*****\n\n");

// process command line arguments
aptr = &args;
GetArgsLoader(argc, argv, aptr);

printf("Build interface is ODBC.\n");

if (aptr->build_index == 0)
  printf("Data load only - no index
creation.\n");
else
  printf("Data load and index
creation.\n");

if (aptr->index_order == 0)
  printf("Clustered indexes will be
created after bulk load.\n");
else
  printf("Clustered indexes will be
created before bulk load.\n");

// set database scale values
if (aptr->scale_down == 1)
{
  printf("*** Scaled Down Database
***\n");
  max_items = MAXITEMS_SCALE_DOWN;
  customers_per_district =
CUSTOMERS_SCALE_DOWN;
  orders_per_district =
ORDERS_SCALE_DOWN;
  first_new_order = 0;
  last_new_order = 30;
}
else
{
  max_items = MAXITEMS;
  customers_per_district =
CUSTOMERS_PER_DISTRICT;
  orders_per_district =
ORDERS_PER_DISTRICT;
  first_new_order = 2100;
  last_new_order = 3000;
}

```

```

    }
    // open connections to SQL Server
    OpenConnections();
    // open file for loader results
    fLoader = fopen(aptr->loader_res_file, "w");
    if (fLoader == NULL)
    {
        printf("Error, loader result file
open failed.");
        exit(-1);
    }
    // start loading data
    sprintf(buffer, "TPC-C load started for %ld
warehouses.\n", aptr->num_warehouses);
    printf("%s", buffer);
    fprintf(fLoader, "%s", buffer);
    main_time_start = (TimeNow() / MILLI);
    // start parallel load threads
    if (aptr->tables_all || aptr->table_item)
    {
        fprintf(fLoader, "\nStarting
loader threads for: item\n");
        hThread[0] = CreateThread(NULL,
            0,
(LPTHREAD_START_ROUTINE) LoadItem,
            NULL,
            0,
            &dwThreadID[0]);
        if (hThread[0] == NULL)
        {
            printf("Error, failed
in creating creating thread = 0.\n");
            exit(-1);
        }
        if (aptr->tables_all || aptr-
>table_warehouse)
        {
            fprintf(fLoader, "Starting loader
threads for: warehouse\n");
            hThread[1] = CreateThread(NULL,

```

```

            0,
(LPTHREAD_START_ROUTINE) LoadWarehouse,
            NULL,
            0,
            &dwThreadID[1]);
        if (hThread[1] == NULL)
        {
            printf("Error, failed
in creating creating thread = 1.\n");
            exit(-1);
        }
        if (aptr->tables_all || aptr-
>table_customer)
        {
            fprintf(fLoader, "Starting loader
threads for: customer\n");
            hThread[2] = CreateThread(NULL,
            0,
(LPTHREAD_START_ROUTINE) LoadCustomer,
            NULL,
            0,
            &dwThreadID[2]);
            if (hThread[2] == NULL)
            {
                printf("Error, failed
in creating creating main thread = 2.\n");
                exit(-1);
            }
            if (aptr->tables_all || aptr->table_orders)
            {
                fprintf(fLoader, "Starting loader
threads for: orders\n");
                hThread[3] = CreateThread(NULL,
            0,
(LPTHREAD_START_ROUTINE) LoadOrders,
            NULL,
            0,

```

```

            &dwThreadID[3]);
            if (hThread[3] == NULL)
            {
                printf("Error, failed
in creating creating main thread = 3.\n");
                exit(-1);
            }
        }
        // Wait for threads to finish...
        for (i=0; i<MAX_MAIN_THREADS; i++)
        {
            if (hThread[i] != NULL)
            {
                WaitForSingleObject(
hThread[i], INFINITE);
                CloseHandle(hThread[i]);
                hThread[i] = NULL;
            }
        }
        main_time_end = (TimeNow() / MILLI);
        sprintf(buffer, "\nTPC-C load completed
successfully in %ld minutes.\n",
            (main_time_end -
main_time_start)/60);
        printf("%s", buffer);
        fprintf(fLoader, "%s", buffer);
        fclose(fLoader);
        SQLFreeEnv(henv);
        exit(0);
        return 0;
    }
}
//=====
//
// Function name: LoadItem
//
//=====
void LoadItem()
{
    long        i_id;
    long        i_im id;
    char        i_name[I_NAME_LEN+1];
    double      i_price;
    char        i_data[I_DATA_LEN+1];
    char        name[20];
    long        time_start;
    RETCODE     rc;
    DBINT       rcint;

```



```

char          bcphint[128];

// Seed with unique number
seed(1);

printf("Loading item table...\n");

// if build index before load
if ((aptr->build_index == 1) && (aptr->index_order == 1))
    BuildIndex("idxitmcl");

InitString(i_name, I_NAME_LEN+1);
InitString(i_data, I_DATA_LEN+1);

sprintf(name, "%s..%s", aptr->database, "item");

rc = bcp_init(i_hdbc1, name, NULL, "logs\\item.err", DB_IN);
if (rc != SUCCEEDED)
    HandleErrorDBC(i_hdbc1);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcphint, "tablock, order (i_id), ROWS_PER_BATCH = 100000");
    rc = bcp_control(i_hdbc1, BCPHINTS, (void*) bcphint);
    if (rc != SUCCEEDED)
        HandleErrorDBC(i_hdbc1);
}

rc = bcp_bind(i_hdbc1, (BYTE *) &i_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
if (rc != SUCCEEDED)
    HandleErrorDBC(i_hdbc1);

rc = bcp_bind(i_hdbc1, (BYTE *) &i_im_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 2);
if (rc != SUCCEEDED)
    HandleErrorDBC(i_hdbc1);

rc = bcp_bind(i_hdbc1, (BYTE *) i_name, 0, I_NAME_LEN, NULL, 0, 0, 3);
if (rc != SUCCEEDED)
    HandleErrorDBC(i_hdbc1);

rc = bcp_bind(i_hdbc1, (BYTE *) &i_price, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 4);
if (rc != SUCCEEDED)
    HandleErrorDBC(i_hdbc1);

rc = bcp_bind(i_hdbc1, (BYTE *) i_data, 0, I_DATA_LEN, NULL, 0, 0, 5);
if (rc != SUCCEEDED)
    HandleErrorDBC(i_hdbc1);

time_start = (TimeNow() / MILLI);

item_rows_loaded = 0;

```

```

for (i_id = 1; i_id <= max_items; i_id++)
{
    i_im_id = RandomNumber(1L, 10000L);

    MakeAlphaString(14, 24, I_NAME_LEN, i_name);

    i_price = ((float) RandomNumber(100L, 10000L))/100.0;

    MakeOriginalAlphaString(26, 50, I_DATA_LEN, i_data, 10);

    rc = bcp_sendrow(i_hdbc1);

    if (rc != SUCCEEDED)
        HandleErrorDBC(i_hdbc1);

    item_rows_loaded++;
    CheckForCommit(i_hdbc1, i_hstmt1, item_rows_loaded, "item", &time_start);
}

rcint = bcp_done(i_hdbc1);
if (rcint < 0)
    HandleErrorDBC(i_hdbc1);

printf("Finished loading item table.\n");

SQLFreeStmt(i_hstmt1, SQL_DROP);
SQLDisconnect(i_hdbc1);
SQLFreeConnect(i_hdbc1);

// if build index after load
if ((aptr->build_index == 1) && (aptr->index_order == 0))
    BuildIndex("idxitmcl");
}

//=====
//
// Function : LoadWarehouse
//
// Loads WAREHOUSE table and loads Stock and District as Warehouses are created
//
//=====

void LoadWarehouse()
{
    short w_id;
    char w_name[W_NAME_LEN+1];
    char w_street_1[ADDRESS_LEN+1];
    char w_street_2[ADDRESS_LEN+1];

```

```

char w_city[ADDRESS_LEN+1];
char w_state[STATE_LEN+1];
char w_zip[ZIP_LEN+1];
double w_tax;
double w_ytd;
char name[20];
long time_start;
RETCODE rc;
DBINT rcint;
char bcphint[128];

// Seed with unique number
seed(2);

printf("Loading warehouse table...\n");

// if build index before load..
if ((aptr->build_index == 1) && (aptr->index_order == 1))
    BuildIndex("idxwarc1");

InitString(w_name, W_NAME_LEN+1);
InitAddress(w_street_1, w_street_2, w_city, w_state, w_zip);

sprintf(name, "%s..%s", aptr->database, "warehouse");

rc = bcp_init(w_hdbc1, name, NULL, "logs\\whouse.err", DB_IN);
if (rc != SUCCEEDED)
    HandleErrorDBC(w_hdbc1);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcphint, "tablock, order (w_id), ROWS_PER_BATCH = %d", aptr->num_warehouses);
    rc = bcp_control(w_hdbc1, BCPHINTS, (void*) bcphint);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);
}

rc = bcp_bind(w_hdbc1, (BYTE *) &w_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 1);
if (rc != SUCCEEDED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_name, 0, W_NAME_LEN, NULL, 0, 0, 2);
if (rc != SUCCEEDED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_street_1, 0, ADDRESS_LEN, NULL, 0, 0, 3);
if (rc != SUCCEEDED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_street_2, 0, ADDRESS_LEN, NULL, 0, 0, 4);

```

```

        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_city, 0,
ADDRESS_LEN, NULL, 0, 0, 5);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_state, 0,
STATE_LEN, NULL, 0, 0, 6);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) w_zip, 0,
ZIP_LEN, NULL, 0, 0, 7);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &w_tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 8);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &w_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        time_start = (TimeNow() / MILLI);

        warehouse_rows_loaded = 0;

        for (w_id = (short)aptr-
>starting_warehouse; w_id <= aptr->num_warehouses;
w_id++)
        {
            MakeAlphaString(6,10, W_NAME_LEN,
w_name);

            MakeAddress(w_street_1,
w_street_2, w_city, w_state, w_zip);

            w_tax = ((float)
RandomNumber(0L,2000L))/10000.00;

            w_ytd = 300000.00;

            rc = bcp_sendrow(w_hdbc1);
            if (rc != SUCCEEDED)

                HandleErrorDBC(w_hdbc1);

            warehouse_rows_loaded++;
            CheckForCommit(w_hdbc1, i_hstmt1,
warehouse_rows_loaded, "warehouse", &time_start);
        }

        rcint = bcp_done(w_hdbc1);
        if (rcint < 0)
            HandleErrorDBC(w_hdbc1);

```

```

        printf("Finished loading warehouse
table.\n");

        // if build index after load...
        if ((aptr->build_index == 1) && (aptr-
>index_order == 0))
            BuildIndex("idxwarc1");

        stock_rows_loaded = 0;
        district_rows_loaded = 0;

        District();
        Stock();
    }

//=====
//
// Function   : District
//
//=====

void District()
{
    short d_id;
    short d_w_id;
    char d_name[D_NAME_LEN+1];
    char d_street_1[ADDRESS_LEN+1];
    char d_street_2[ADDRESS_LEN+1];
    char d_city[ADDRESS_LEN+1];
    char d_state[STATE_LEN+1];
    char d_zip[ZIP_LEN+1];
    double d_tax;
    double d_ytd;
    char name[20];
    long d_next_o_id;
    long time_start;
    int w_id;
    RETCODE rc;
    DBINT rcint;
    char bcpint[128];

    // Seed with unique number
    seed(4);

    printf("Loading district table...\n");

    // build index before load
    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        BuildIndex("idxdiscl");

    InitString(d_name, D_NAME_LEN+1);
    InitAddress(d_street_1, d_street_2, d_city,
d_state, d_zip);
    sprintf(name, "%s.%s", aptr->database,
"district");

    rc = bcp_init(w_hdbc1, name, NULL,
"logs\\district.err", DB_IN);

```

```

        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        {
            sprintf(bcpint, "tablock, order
(d_w_id, d_id), ROWS_PER_BATCH = %u", (aptr-
>num_warehouses * 10));
            rc = bcp_control(w_hdbc1,
BCPHINTS, (void*) bcpint);
            if (rc != SUCCEEDED)

                HandleErrorDBC(w_hdbc1);
        }

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 1);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_name, 0,
D_NAME_LEN, NULL, 0, 0, 3);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_street_1,
0, ADDRESS_LEN, NULL, 0, 0, 4);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_street_2,
0, ADDRESS_LEN, NULL, 0, 0, 5);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_city, 0,
ADDRESS_LEN, NULL, 0, 0, 6);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_state, 0,
STATE_LEN, NULL, 0, 0, 7);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_zip, 0,
ZIP_LEN, NULL, 0, 0, 8);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_tax, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 10);

```

```

    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *)
&d_next_o_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4,
11);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    d_ytd = 30000.0;

    d_next_o_id = orders_per_district+1;

    time_start = (TimeNow() / MILLI);

    for (w_id = aptr->starting_warehouse; w_id
<= aptr->num_warehouses; w_id++)
    {
        d_w_id = w_id;

        for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
        {
            MakeAlphaString(6,10,D_NAME_LEN, d_name);

            MakeAddress(d_street_1,
d_street_2, d_city, d_state, d_zip);

            d_tax = ((float)
RandomNumber(0L,2000L))/10000.00;

            rc =
bcp_sendrow(w_hdbc1);

            if (rc != SUCCEEDED)

                HandleErrorDBC(w_hdbc1);

            district_rows_loaded++;
            CheckForCommit(w_hdbc1,
&w_hstmt1, district_rows_loaded, "district",
&time_start);
        }
    }

    rcint = bcp_done(w_hdbc1);
    if (rcint < 0)
        HandleErrorDBC(w_hdbc1);

    printf("Finished loading district
table.\n");

    // if build index after load...
    if ((aptr->build_index == 1) && (aptr-
>index_order == 0))
        BuildIndex("idxdiscl");

    return;
}

```

```

//=====
//
// Function   : Stock
//
//=====

void Stock()
{
    long s_i_id;
    short s_w_id;
    short s_quantity;
    char s_dist_01[S_DIST_LEN+1];
    char s_dist_02[S_DIST_LEN+1];
    char s_dist_03[S_DIST_LEN+1];
    char s_dist_04[S_DIST_LEN+1];
    char s_dist_05[S_DIST_LEN+1];
    char s_dist_06[S_DIST_LEN+1];
    char s_dist_07[S_DIST_LEN+1];
    char s_dist_08[S_DIST_LEN+1];
    char s_dist_09[S_DIST_LEN+1];
    char s_dist_10[S_DIST_LEN+1];
    long s_ytd;
    short s_order_cnt;
    short s_remote_cnt;
    char s_data[S_DATA_LEN+1];
    short len;
    char name[20];
    long time_start;
    RETCODE rc;
    DBINT rcint;
    char bcpint[128];

    // Seed with unique number
    seed(3);

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        BuildIndex("idxstck1");

    sprintf(name, "%s.%s", aptr->database,
"stock");

    rc = bcp_init(w_hdbc1, name, NULL,
"logs\\stock.err", DB IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
    {
        sprintf(bcpint, "tablock, order
(s_i_id, s_w_id), ROWS PER BATCH = %u", (aptr-
>num_warehouses * 10000));
        rc = bcp_control(w_hdbc1,
BCPHINTS, (void*) bcpint);
        if (rc != SUCCEEDED)

            HandleErrorDBC(w_hdbc1);
    }
}

```

```

    rc = bcp_bind(w_hdbc1, (BYTE *) &s_i_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    bcp_bind(w_hdbc1, (BYTE *) &s_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *)
&s_quantity, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2,
3);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_01,
0, S_DIST_LEN, NULL, 0, 0, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_02,
0, S_DIST_LEN, NULL, 0, 0, 5);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_03,
0, S_DIST_LEN, NULL, 0, 0, 6);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_04,
0, S_DIST_LEN, NULL, 0, 0, 7);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_05,
0, S_DIST_LEN, NULL, 0, 0, 8);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_06,
0, S_DIST_LEN, NULL, 0, 0, 9);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_07,
0, S_DIST_LEN, NULL, 0, 0, 10);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_08,
0, S_DIST_LEN, NULL, 0, 0, 11);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_09,
0, S_DIST_LEN, NULL, 0, 0, 12);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_10,
0, S_DIST_LEN, NULL, 0, 0, 13);

```

```

        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 14);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *)
&s_order_cnt, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2,
15);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *)
&s_remote_cnt, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2,
16);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_data, 0,
S_DATA_LEN, NULL, 0, 0, 17);
        if (rc != SUCCEED)
            HandleErrorDBC(w_hdbc1);

        s_ytd = s_order_cnt = s_remote_cnt = 0;

        time_start = (TimeNow() / MILLI);

        printf("...Loading stock table\n");

        for (s_i_id=1; s_i_id <= max_items;
s_i_id++)
        {
            for (s_w_id = (short)aptr-
>starting_warehouse; s_w_id <= aptr->num_warehouses;
s_w_id++)
            {
                s_quantity =
(short)RandomNumber(10L,100L);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_01);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_02);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_03);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_04);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_05);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_06);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_07);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_08);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_09);
                len =
                MakeAlphaString(24,24,S_DIST_LEN, s_dist_10);

```

```

                len =
                MakeOriginalAlphaString(26,50, S_DATA_LEN,
s_data,10);
                rc =
                bcp_sendrow(w_hdbc1);
                if (rc != SUCCEED)
                    HandleErrorDBC(w_hdbc1);

                stock_rows_loaded++;
                CheckForCommit(w_hdbc1,
w_hstmt1, stock_rows_loaded, "stock", &time_start);
            }
        }

        rcint = bcp_done(w_hdbc1);
        if (rcint < 0)
            HandleErrorDBC(w_hdbc1);

        printf("Finished loading stock table.\n");

        SQLFreeStmt(w_hstmt1, SQL_DROP);
        SQLDisconnect(w_hdbc1);
        SQLFreeConnect(w_hdbc1);

        // if build index after load...
        if ((aptr->build_index == 1) && (aptr-
>index_order == 0))
            BuildIndex("idxstkl");

        return;
    }

//=====
//
// Function   : LoadCustomer
//
//=====

void LoadCustomer()
{
    LOADER_TIME_STRUCT
customer_time_start;
    LOADER_TIME_STRUCT      history_time_start;
    short                    w_id;
    short                    d_id;
    DWORD
dwThreadId[MAX_CUSTOMER_THREADS];
    HANDLE
hThread[MAX_CUSTOMER_THREADS];
    char                      name[20];
    RETCODE
rc;
    DBINT
rcint;
    char
bcphint[128];

```

```

    char
cmd[256];
    // SQLRETURN
rc_l;
    // SQLSMALLINT
recnum, MsgLen;
    // SQLCHAR
        SqlState[6],
Msg[SQL_MAX_MESSAGE_LENGTH];
    // SQLINTEGER
NativeError;

    // Seed with unique number
seed(5);

    printf("Loading customer and history
tables...\n");

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        BuildIndex("idxcuscl");

    // Initialize bulk copy
    sprintf(name, "%s..%s", aptr->database,
"customer");

    rc = bcp_init(c_hdbc1, name, NULL,
"logs\\customer.err", DB_IN);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc1);

    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
    {
        sprintf(bcphint, "tablock, order
(c_w_id, c_d_id, c_id), ROWS_PER_BATCH = %u", (aptr-
>num_warehouses * 30000));
        rc = bcp_control(c_hdbc1,
BCPHINTS, (void*) bcphint);
        if (rc != SUCCEED)
            HandleErrorDBC(c_hdbc1);
    }

    sprintf(name, "%s..%s", aptr->database,
"history");

    rc = bcp_init(c_hdbc2, name, NULL,
"logs\\history.err", DB_IN);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    sprintf(bcphint, "tablock");
    rc = bcp_control(c_hdbc2, BCPHINTS, (void*)
bcphint);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    customer_rows_loaded = 0;
    history_rows_loaded = 0;

    CustomerBufinit();

```

```

        customer_time_start.time_start = (TimeNow()
/ MILLI);
    history_time_start.time_start = (TimeNow()
/ MILLI);
    for (w_id = (short)aptr-
>starting_warehouse; w_id <= aptr->num_warehouses;
w_id++)
    {
        for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
        {
            CustomerBufLoad(d_id,
w_id);
loading threads here...
thread
        printf("...Loading
customer table for: d_id = %d, w_id = %d\n", d_id,
w_id);
        CreateThread(NULL,
(LPTHREAD_START_ROUTINE) LoadCustomerTable,
&customer_time_start,
&dwThreadID[0]);
        if (hThread[0] == NULL)
        {
            printf("Error, failed in creating creating
thread = 0.\n");
            exit(-1);
        }
        // Start History table
thread
        printf("...Loading
history table for: d_id = %d, w_id = %d\n", d_id,
w_id);
        CreateThread(NULL,
(LPTHREAD_START_ROUTINE) LoadHistoryTable,
&history_time_start,
0,
&dwThreadID[1]);
        if (hThread[1] == NULL)
        {
            printf("Error, failed in creating creating
thread = 1.\n");
            exit(-1);
        }
        WaitForSingleObject(
hThread[0], INFINITE );
        WaitForSingleObject(
hThread[1], INFINITE );
        if
(CloseHandle(hThread[0]) == FALSE)
        {
            printf("Error, failed in closing customer
thread handle with errno: %d\n", GetLastError());
        }
        if
(CloseHandle(hThread[1]) == FALSE)
        {
            printf("Error, failed in closing history
thread handle with errno: %d\n", GetLastError());
        }
        }
        // flush the bulk connection
rcint = bcp_done(c_hdbc1);
if (rcint < 0)
    HandleErrorDBC(c_hdbc1);
rcint = bcp_done(c_hdbc2);
if (rcint < 0)
    HandleErrorDBC(c_hdbc2);
        printf("Finished loading customer
table.\n");
        // if build index after load...
if ((aptr->build_index == 1) && (aptr-
>index_order == 0))
    BuildIndex("idxcuscl");
// build non-clustered index
if (aptr->build_index == 1)
    BuildIndex("idxcusnc");
// Output the NURAND used for the loader
into C_FIRST for C_ID = 1,
// C_W_ID = 1, and C_D_ID = 1
sprintf(cmd, "isql -S%s -U%s -P%s -d%s -e -
Q\"update customer set c_first = 'C_LOAD = %d' where
c_id = 1 and c_w_id = 1 and c_d_id = 1\" >
logs\\nurand_load.log",
aptr->server,
aptr->user,
aptr-
>password,
aptr-
>database,
LOADER_NURAND_C);
system(cmd);
SQLFreeStmt(c_hstmt1, SQL_DROP);
SQLDisconnect(c_hdbc1);
SQLFreeConnect(c_hdbc1);
SQLFreeStmt(c_hstmt2, SQL_DROP);
SQLDisconnect(c_hdbc2);
SQLFreeConnect(c_hdbc2);
return;
}
//=====
//
// Function : CustomerBufInit
//=====
void CustomerBufInit()
{
    int i;
    for (i=0;i<customers_per_district;i++)
    {
        customer_buf[i].c_id = 0;
        customer_buf[i].c_d_id = 0;
        customer_buf[i].c_w_id = 0;
        strcpy(customer_buf[i].c_first,"");
        strcpy(customer_buf[i].c_middle,"");
        strcpy(customer_buf[i].c_last,"");
    }
}

```

```

(LPTHREAD_START_ROUTINE) LoadHistoryTable,
&history_time_start,
0,
&dwThreadID[1]);
        if (hThread[1] == NULL)
        {
            printf("Error, failed in creating creating
thread = 1.\n");
            exit(-1);
        }
        WaitForSingleObject(
hThread[0], INFINITE );
        WaitForSingleObject(
hThread[1], INFINITE );
        if
(CloseHandle(hThread[0]) == FALSE)
        {
            printf("Error, failed in closing customer
thread handle with errno: %d\n", GetLastError());
        }
        if
(CloseHandle(hThread[1]) == FALSE)
        {
            printf("Error, failed in closing history
thread handle with errno: %d\n", GetLastError());
        }
        }
        // flush the bulk connection
rcint = bcp_done(c_hdbc1);
if (rcint < 0)
    HandleErrorDBC(c_hdbc1);
rcint = bcp_done(c_hdbc2);
if (rcint < 0)
    HandleErrorDBC(c_hdbc2);
        printf("Finished loading customer
table.\n");
        // if build index after load...
if ((aptr->build_index == 1) && (aptr-
>index_order == 0))
    BuildIndex("idxcuscl");
// build non-clustered index
if (aptr->build_index == 1)
    BuildIndex("idxcusnc");
// Output the NURAND used for the loader
into C_FIRST for C_ID = 1,
// C_W_ID = 1, and C_D_ID = 1
sprintf(cmd, "isql -S%s -U%s -P%s -d%s -e -
Q\"update customer set c_first = 'C_LOAD = %d' where
c_id = 1 and c_w_id = 1 and c_d_id = 1\" >
logs\\nurand_load.log",
aptr->server,
aptr->user,
aptr-
>password,
aptr-
>database,
LOADER_NURAND_C);
system(cmd);
SQLFreeStmt(c_hstmt1, SQL_DROP);
SQLDisconnect(c_hdbc1);
SQLFreeConnect(c_hdbc1);
SQLFreeStmt(c_hstmt2, SQL_DROP);
SQLDisconnect(c_hdbc2);
SQLFreeConnect(c_hdbc2);
return;
}
//=====
//
// Function : CustomerBufInit
//=====
void CustomerBufInit()
{
    int i;
    for (i=0;i<customers_per_district;i++)
    {
        customer_buf[i].c_id = 0;
        customer_buf[i].c_d_id = 0;
        customer_buf[i].c_w_id = 0;
        strcpy(customer_buf[i].c_first,"");
        strcpy(customer_buf[i].c_middle,"");
        strcpy(customer_buf[i].c_last,"");
    }
}

```

```

(LPTHREAD_START_ROUTINE) LoadHistoryTable,
&history_time_start,
0,
&dwThreadID[1]);
        if (hThread[1] == NULL)
        {
            printf("Error, failed in creating creating
thread = 1.\n");
            exit(-1);
        }
        WaitForSingleObject(
hThread[0], INFINITE );
        WaitForSingleObject(
hThread[1], INFINITE );
        if
(CloseHandle(hThread[0]) == FALSE)
        {
            printf("Error, failed in closing customer
thread handle with errno: %d\n", GetLastError());
        }
        if
(CloseHandle(hThread[1]) == FALSE)
        {
            printf("Error, failed in closing history
thread handle with errno: %d\n", GetLastError());
        }
        }
        // flush the bulk connection
rcint = bcp_done(c_hdbc1);
if (rcint < 0)
    HandleErrorDBC(c_hdbc1);
rcint = bcp_done(c_hdbc2);
if (rcint < 0)
    HandleErrorDBC(c_hdbc2);
        printf("Finished loading customer
table.\n");
        // if build index after load...
if ((aptr->build_index == 1) && (aptr-
>index_order == 0))
    BuildIndex("idxcuscl");
// build non-clustered index
if (aptr->build_index == 1)
    BuildIndex("idxcusnc");
// Output the NURAND used for the loader
into C_FIRST for C_ID = 1,
// C_W_ID = 1, and C_D_ID = 1
sprintf(cmd, "isql -S%s -U%s -P%s -d%s -e -
Q\"update customer set c_first = 'C_LOAD = %d' where
c_id = 1 and c_w_id = 1 and c_d_id = 1\" >
logs\\nurand_load.log",
aptr->server,
aptr->user,
aptr-
>password,
aptr-
>database,
LOADER_NURAND_C);
system(cmd);
SQLFreeStmt(c_hstmt1, SQL_DROP);
SQLDisconnect(c_hdbc1);
SQLFreeConnect(c_hdbc1);
SQLFreeStmt(c_hstmt2, SQL_DROP);
SQLDisconnect(c_hdbc2);
SQLFreeConnect(c_hdbc2);
return;
}
//=====
//
// Function : CustomerBufInit
//=====
void CustomerBufInit()
{
    int i;
    for (i=0;i<customers_per_district;i++)
    {
        customer_buf[i].c_id = 0;
        customer_buf[i].c_d_id = 0;
        customer_buf[i].c_w_id = 0;
        strcpy(customer_buf[i].c_first,"");
        strcpy(customer_buf[i].c_middle,"");
        strcpy(customer_buf[i].c_last,"");
    }
}

```

```

strcpy(customer_buf[i].c_street_1,"");
strcpy(customer_buf[i].c_street_2,"");
strcpy(customer_buf[i].c_city,"");
strcpy(customer_buf[i].c_state,"");
strcpy(customer_buf[i].c_zip,"");
strcpy(customer_buf[i].c_phone,"");
strcpy(customer_buf[i].c_credit,"");

customer_buf[i].c_credit_lim = 0;
customer_buf[i].c_discount =
(float) 0;

// fix to avoid ODBC float to
numeric conversion problem.
//
customer_buf[i].c_balance = 0;
strcpy(customer_buf[i].c_balance,"");

customer_buf[i].c_ytd_payment =
0;
customer_buf[i].c_payment_cnt =
0;
customer_buf[i].c_delivery_cnt =
0;

strcpy(customer_buf[i].c_data,"");

customer_buf[i].h_amount = 0;

strcpy(customer_buf[i].h_data,"");
}

//=====
//
// Function : CustomerBufLoad
//
// Fills shared buffer for HISTORY and CUSTOMER
//=====
void CustomerBufLoad(int d_id, int w_id)
{
long i;
CUSTOMER_SORT_STRUCT
c[CUSTOMERS_PER_DISTRICT];

```

```

for (i=0;i<customers_per_district;i++)
{
if (i < 1000)
LastName(i,
c[i].c_last);
else
LastName(NURand(255,0,999,LOADER_NURAND_C),
c[i].c_last);

MakeAlphaString(8,16,FIRST_NAME_LEN,
c[i].c_first);

c[i].c_id = i+1;
}

printf("...Loading customer buffer for:
d_id = %d, w_id = %d\n",
d_id, w_id);

for (i=0;i<customers_per_district;i++)
{
customer_buf[i].c_d_id = d_id;
customer_buf[i].c_w_id = w_id;
customer_buf[i].h_amount = 10.0;

customer_buf[i].c_ytd_payment =
10.0;
customer_buf[i].c_payment_cnt =
1;
customer_buf[i].c_delivery_cnt =
0;

// Generate CUSTOMER and HISTORY
data

customer_buf[i].c_id = c[i].c_id;
strcpy(customer_buf[i].c_first,
c[i].c_first);
strcpy(customer_buf[i].c_last,
c[i].c_last);

customer_buf[i].c_middle[0] =
'o';
customer_buf[i].c_middle[1] =
'E';

MakeAddress(customer_buf[i].c_street_1,
customer_buf[i].c_street_2,
customer_buf[i].c_city,
customer_buf[i].c_state,

```

```

customer_buf[i].c_zip);

MakeNumberString(16, 16,
PHONE_LEN, customer_buf[i].c_phone);

if (RandomNumber(1L, 100L) > 10)
customer_buf[i].c_credit[0] = 'G';
else
customer_buf[i].c_credit[0] = 'B';
customer_buf[i].c_credit[1] =
'C';

customer_buf[i].c_credit_lim =
50000.0;
customer_buf[i].c_discount =
((float) RandomNumber(0L, 5000L) / 10000.0;

// fix to avoid ODBC float to
numeric conversion problem.
// customer_buf[i].c_balance = -
10.0;

strcpy(customer_buf[i].c_balance,"-10.0");

MakeAlphaString(500, 500,
C_DATA_LEN, customer_buf[i].c_data);

// Generate HISTORY data
MakeAlphaString(12, 24,
H_DATA_LEN, customer_buf[i].h_data);
}

//=====
//
// Function : LoadCustomerTable
//
//=====
void LoadCustomerTable(LOADER_TIME_STRUCT
*customer_time_start)
{
int i;
long c_id;
short c_d_id;
short c_w_id;
char c_first[FIRST_NAME_LEN+1];
char c_middle[MIDDLE_NAME_LEN+1];
char c_last[LAST_NAME_LEN+1];
char c_street_1[ADDRESS_LEN+1];
char c_street_2[ADDRESS_LEN+1];
char c_city[ADDRESS_LEN+1];
char c_state[STATE_LEN+1];
char c_zip[ZIP_LEN+1];
char c_phone[PHONE_LEN+1];
char c_credit[CREDIT_LEN+1];

```

```

double    c_credit_lim;
double    c_discount;

    // fix to avoid ODBC float to numeric
conversion problem.
    // double                c_balance;
char      c_balance[6];

double    c_ytd_payment;
short     c_payment_cnt;
short     c_delivery_cnt;
char      c_data[C_DATA_LEN+1];
char      c_since[C_SINCE_LEN+1];

RETCODE   rc;

rc = bcp_bind(c_hdbc1, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_first, 0,
FIRST_NAME_LEN, NULL, 0, 0, 4);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_middle, 0,
MIDDLE_NAME_LEN, NULL, 0, 0, 5);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_last, 0,
LAST_NAME_LEN, NULL, 0, 0, 6);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_street_1, 0,
ADDRESS_LEN, NULL, 0, 0, 7);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_street_2, 0,
ADDRESS_LEN, NULL, 0, 0, 8);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_city, 0,
ADDRESS_LEN, NULL, 0, 0, 9);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_state, 0,
STATE_LEN, NULL, 0, 0, 10);

```

```

if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_zip, 0,
ZIP_LEN, NULL, 0, 0, 11);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_phone, 0,
PHONE_LEN, NULL, 0, 0, 12);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_since,
0, C_SINCE_LEN, NULL, 0, SQLCHARACTER, 13);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) c_credit, 0,
CREDIT_LEN, NULL, 0, 0, 14);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_credit_lim, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 15);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_discount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 16);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

    // fix to avoid ODBC float to numeric
conversion problem.
    // rc = bcp_bind(c_hdbc1, (BYTE *) &c_balance,
0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 17);
    // if (rc != SUCCEEDED)
    //     HandleErrorDBC(c_hdbc1);
rc = bcp_bind(c_hdbc1, (BYTE *) c_balance, 0, 5,
NULL, 0, SQLCHARACTER, 17);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_ytd_payment,
0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 18);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *) &c_payment_cnt,
0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 19);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

rc = bcp_bind(c_hdbc1, (BYTE *)
&c_delivery_cnt, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2,
20);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

```

```

rc = bcp_bind(c_hdbc1, (BYTE *) c_data, 0, 500,
NULL, 0, 0, 21);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);

for (i = 0; i < customers_per_district; i++)
{
    c_id = customer_buf[i].c_id;
    c_d_id = customer_buf[i].c_d_id;
    c_w_id = customer_buf[i].c_w_id;

    strcpy(c_first,
customer_buf[i].c_first);
    strcpy(c_middle,
customer_buf[i].c_middle);
    strcpy(c_last,
customer_buf[i].c_last);
    strcpy(c_street_1,
customer_buf[i].c_street_1);
    strcpy(c_street_2,
customer_buf[i].c_street_2);
    strcpy(c_city,
customer_buf[i].c_city);
    strcpy(c_state,
customer_buf[i].c_state);
    strcpy(c_zip,
customer_buf[i].c_zip);
    strcpy(c_phone,
customer_buf[i].c_phone);
    strcpy(c_credit,
customer_buf[i].c_credit);

    FormatDate(&c_since);

    c_credit_lim =
customer_buf[i].c_credit_lim;
    c_discount =
customer_buf[i].c_discount;

    // fix to avoid ODBC float to
numeric conversion problem.
    // c_balance =
customer_buf[i].c_balance;
    strcpy(c_balance,
customer_buf[i].c_balance);

    c_ytd_payment =
customer_buf[i].c_ytd_payment;
    c_payment_cnt =
customer_buf[i].c_payment_cnt;
    c_delivery_cnt =
customer_buf[i].c_delivery_cnt;

    strcpy(c_data,
customer_buf[i].c_data);

    // Send data to server
rc = bcp_sendrow(c_hdbc1);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);
}

```

```

        customer_rows_loaded++;
        CheckForCommit(c_hdbc1, c_hstmt1,
customer_rows_loaded, "customer",
&customer_time_start->time_start);
    }

}

//=====
//
// Function   : LoadHistoryTable
//
//=====
void LoadHistoryTable(LOADER_TIME_STRUCT
*history_time_start)
{
    int         i;
    long        c_id;
    short       c_d_id;
    short       c_w_id;
    double      h_amount;
    char        h_data[H_DATA_LEN+1];
    char        h_date[H_DATE_LEN+1];

    RETCODE     rc;

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 5);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &h_date, 0,
H_DATE_LEN, NULL, 0, SQLCHARACTER, 6);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &h_amount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 7);
    if (rc != SUCCEEDED)

```

```

        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) h_data, 0,
H_DATA_LEN, NULL, 0, 0, 8);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

    for (i = 0; i < customers_per_district; i++)
    {
        c_id = customer_buf[i].c_id;
        c_d_id = customer_buf[i].c_d_id;
        c_w_id = customer_buf[i].c_w_id;
        h_amount =
customer_buf[i].h_amount;
        strcpy(h_data,
customer_buf[i].h_data);

        FormatDate(&h_date);

        // send to server
        rc = bcp_sendrow(c_hdbc2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc2);

        history_rows_loaded++;
        CheckForCommit(c_hdbc2, c_hstmt2,
history_rows_loaded, "history", &history_time_start-
>time_start);
    }

}

//=====
//
// Function   : LoadOrders
//
//=====
void LoadOrders()
{
    LOADER_TIME_STRUCT    orders_time_start;
    LOADER_TIME_STRUCT    new_order_time_start;
    LOADER_TIME_STRUCT    order_line_time_start;
    short                 short         w_id;
    short                 short         d_id;
    DWORD                 dwThreadID[MAX_ORDER_THREADS];
    HANDLE                hThread[MAX_ORDER_THREADS];
    char                  name[20];
    RETCODE               rc;
    char                  bcphint[128];

```

```

    // seed with unique number
    seed(6);

    printf("Loading orders...\n");

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
    {
        BuildIndex("idxordcl");
        BuildIndex("idxnodcl");
        BuildIndex("idxodlcl");
    }

    // initialize bulk copy
    sprintf(name, "%s.%s", aptr->database,
"orders");

    rc = bcp_init(o_hdbc1, name, NULL,
"logs\\orders.err", DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
    {
        sprintf(bcphint, "tablock, order
(o_w_id, o_d_id, o_id), ROWS_PER_BATCH = %u", (aptr-
>num_warehouses * 30000));
        rc = bcp_control(o_hdbc1,
BCPHINTS, (void*) bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc1);
    }

    sprintf(name, "%s.%s", aptr->database,
"new_order");

    rc = bcp_init(o_hdbc2, name, NULL,
"logs\\neword.err", DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
    {
        sprintf(bcphint, "tablock, order
(no_w_id, no_d_id, no_o_id), ROWS_PER_BATCH = %u",
(aptr->num_warehouses * 9000));
        rc = bcp_control(o_hdbc2,
BCPHINTS, (void*) bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc2);
    }

    sprintf(name, "%s.%s", aptr->database,
"order_line");

    rc = bcp_init(o_hdbc3, name, NULL,
"logs\\ordline.err", DB_IN);

```



```

        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc3);

        if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        {
            sprintf(bcphint, "tablock, order
(ol_w_id, ol_d_id, ol_o_id, ol_number),
ROWS_PER_BATCH = %u", (aptr->num_warehouses *
300000));
            rc = bcp_control(o_hdbc3,
BCPHINTS, (void*) bcphint);
            if (rc != SUCCEEDED)

                HandleErrorDBC(o_hdbc3);
        }

        orders_rows_loaded = 0;
        new_order_rows_loaded = 0;
        order_line_rows_loaded = 0;

        OrdersBufInit();

        orders_time_start.time_start = (TimeNow() /
MILLI);
        new_order_time_start.time_start =
(TimeNow() / MILLI);
        order_line_time_start.time_start =
(TimeNow() / MILLI);

        for (w_id = (short)aptr-
>starting_warehouse; w_id <= aptr->num_warehouses;
w_id++)
        {
            for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
            {
                OrdersBufLoad(d_id,
w_id);

                // start parallel
                loading threads here...

                // start Orders table
                thread

                printf("...Loading
Order Table for: d_id = %d, w_id = %d\n", d_id,
w_id);

                hThread[0] =
CreateThread(NULL,

                    0,

                    (LPTHREAD_START_ROUTINE) LoadOrdersTable,

                    &orders_time_start,

                    0,

```

```

&dwThreadID[0]);

                if (hThread[0] == NULL)
                {
                    printf("Error, failed in creating creating
thread = 0.\n");
                    exit(-1);
                }

                // start NewOrder table
                thread

                printf("...Loading New-
Order Table for: d_id = %d, w_id = %d\n", d_id,
w_id);

                hThread[1] =
CreateThread(NULL,

                    0,

                    (LPTHREAD_START_ROUTINE) LoadNewOrderTable,

                    &new_order_time_start,

                    0,

                    &dwThreadID[1]);

                if (hThread[1] == NULL)
                {
                    printf("Error, failed in creating creating
thread = 1.\n");
                    exit(-1);
                }

                // start Order-Line
                table thread

                printf("...Loading
Order-Line Table for: d_id = %d, w_id = %d\n", d_id,
w_id);

                hThread[2] =
CreateThread(NULL,

                    0,

                    (LPTHREAD_START_ROUTINE) LoadOrderLineTable,

                    &order_line_time_start,

                    0,

```

```

&dwThreadID[2]);

                if (hThread[2] == NULL)
                {
                    printf("Error, failed in creating creating
thread = 2.\n");
                    exit(-1);
                }

                WaitForSingleObject(
hThread[0], INFINITE );
                WaitForSingleObject(
hThread[1], INFINITE );
                WaitForSingleObject(
hThread[2], INFINITE );

                if
(CloseHandle(hThread[0]) == FALSE)
                {
                    printf("Error, failed in closing Orders
thread handle with errno: %d\n", GetLastError());
                }

                if
(CloseHandle(hThread[1]) == FALSE)
                {
                    printf("Error, failed in closing NewOrder
thread handle with errno: %d\n", GetLastError());
                }

                if
(CloseHandle(hThread[2]) == FALSE)
                {
                    printf("Error, failed in closing OrderLine
thread handle with errno: %d\n", GetLastError());
                }
            }

            printf("Finished loading orders.\n");

            return;
        }

//=====
//
// Function : OrdersBufInit
//
// Clears shared buffer for ORDERS, NEWORDER, and
ORDERLINE
//
//=====

```

```

void OrdersBufInit()
{
    int i;
    int j;

    for (i=0;i<orders_per_district;i++)
    {
        orders_buf[i].o_id = 0;
        orders_buf[i].o_d_id = 0;
        orders_buf[i].o_w_id = 0;
        orders_buf[i].o_c_id = 0;
        orders_buf[i].o_carrier_id = 0;
        orders_buf[i].o_ol_cnt = 0;
        orders_buf[i].o_all_local = 0;

        for (j=0;j<=14;j++)
        {
            orders_buf[i].o_ol[j].ol = 0;

            orders_buf[i].o_ol[j].ol_i_id = 0;

            orders_buf[i].o_ol[j].ol_supply_w_id = 0;

            orders_buf[i].o_ol[j].ol_quantity = 0;

            orders_buf[i].o_ol[j].ol_amount = 0;

            strcpy(orders_buf[i].o_ol[j].ol_dist_info,"
");
        }
    }

}

//=====
//
// Function : OrdersBufLoad
//
// Fills shared buffer for ORDERS, NEWORDER, and
// ORDERLINE
//
//=====
void OrdersBufLoad(int d_id, int w_id)
{
    int cust[ORDERS_PER_DIST+1];
    long o_id;
    short ol;

    printf("...Loading Order Buffer for: d_id =
%d, w_id = %d\n",
            d_id, w_id);

    GetPermutation(cust, ORDERS_PER_DIST);
}

```

```

for
(o_id=0;o_id<orders_per_district;o_id++)
{
    data // Generate ORDER and NEW-ORDER

        orders_buf[o_id].o_d_id = d_id;
        orders_buf[o_id].o_w_id = w_id;
        orders_buf[o_id].o_id = o_id+1;
        orders_buf[o_id].o_c_id =

cust[o_id+1];
        orders_buf[o_id].o_ol_cnt =
(short)RandomNumber(5L, 15L);

        if (o_id < first_new_order)
        {
            orders_buf[o_id].o_carrier_id =
(short)RandomNumber(1L, 10L);

            orders_buf[o_id].o_all_local = 1;
        }
        else
        {
            orders_buf[o_id].o_carrier_id = 0;
            orders_buf[o_id].o_all_local = 1;
        }

        for (ol=0;
ol<orders_buf[o_id].o_ol_cnt; ol++)
        {

            orders_buf[o_id].o_ol[ol].ol = ol+1;

            orders_buf[o_id].o_ol[ol].ol_i_id =
RandomNumber(1L, max_items);

            orders_buf[o_id].o_ol[ol].ol_supply_w_id =
w_id;

            orders_buf[o_id].o_ol[ol].ol_quantity = 5;
            MakeAlphaString(24, 24,
OL_DIST_INFO_LEN,
&orders_buf[o_id].o_ol[ol].ol_dist_info);

        data // Generate ORDER-LINE

            first_new_order) if (o_id <

                {
                    orders_buf[o_id].o_ol[ol].ol_amount = 0;
                    // Added to
insure ol_delivery_d set properly during load

                    FormatDate(&orders_buf[o_id].o_ol[ol].ol_de
livery_d);
                }
            }
}

```

```

}
else
{
    orders_buf[o_id].o_ol[ol].ol_amount =
RandomNumber(1,999999)/100.0;
    // Added to
insure ol_delivery_d set properly during load

    // odbc
    datetime format

        strcpy(orders_buf[o_id].o_ol[ol].ol_deliver
y_d,"1899-12-31 12:00:00.000");
    }
}

}

//=====
//
// Function : LoadOrdersTable
//
//=====
void LoadOrdersTable(LOADER_TIME_STRUCT
*orders_time_start)
{
    int i;

    long o_id;
    short o_d_id;
    short o_w_id;

    long o_c_id;
    short o_carrier_id;
    short o_ol_cnt;
    short o_all_local;

    char
o_entry_d[O_ENTRY_D_LEN+1];
    RETCODE rc;
    DBINT rcint;

    // bind ORDER data
    rc = bcp_bind(o_hdbc1, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);
}

```

```

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_entry_d,
0, O_ENTRY_D_LEN, NULL, 0, SQLCHARACTER, 5);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_carrier_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 6);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_ol_cnt, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 7);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_all_local, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 8);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    for (i = 0; i < orders_per_district; i++)
    {
        o_id =
orders_buf[i].o_id;
        o_d_id =
orders_buf[i].o_d_id;
        o_w_id =
orders_buf[i].o_w_id;
        o_c_id =
orders_buf[i].o_c_id;
        o_carrier_id =
orders_buf[i].o_carrier_id;
        o_ol_cnt =
orders_buf[i].o_ol_cnt;
        o_all_local =
orders_buf[i].o_all_local;

        FormatDate(&o_entry_d);

        // send data to server
        rc = bcp_sendrow(o_hdbc1);
        if (rc != SUCCEEDED)

            HandleErrorDBC(o_hdbc1);

        orders_rows_loaded++;

        CheckForCommit(o_hdbc1, o_hstmt1,
orders_rows_loaded, "orders", &orders_time_start-
>time_start);
    }

    // rcint = bcp_batch(o_hdbc1);
    // if (rcint < 0)
    //     HandleErrorDBC(o_hdbc1);

```

```

        if ((o_w_id == aptr->num_warehouses) &&
(o_d_id == 10))
        {
            rcint = bcp_done(o_hdbc1);

            if (rcint < 0)

                HandleErrorDBC(o_hdbc1);

            SQLFreeStmt(o_hstmt1, SQL_DROP);
            SQLDisconnect(o_hdbc1);
            SQLFreeConnect(o_hdbc1);

            // if build index after load...
            if ((aptr->build_index == 1) &&
(aptr->index_order == 0))
                BuildIndex("idxordc1");

            // build non-clustered index
            if (aptr->build_index == 1)
                BuildIndex("idxordnc");
        }
    }

//=====
//
// Function : LoadNewOrderTable
//
//=====

void LoadNewOrderTable(LOADER_TIME_STRUCT
*new_order_time_start)
{
    int i;
    long o_id;
    short o_d_id;
    short o_w_id;

    RETCODE rc;
    DBINT rcint;

    // Bind NEW-ORDER data

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);

    for (i = first_new_order; i <
last_new_order; i++)

```

```

    {
        o_id = orders_buf[i].o_id;
        o_d_id = orders_buf[i].o_d_id;
        o_w_id = orders_buf[i].o_w_id;

        rc = bcp_sendrow(o_hdbc2);
        if (rc != SUCCEEDED)

            HandleErrorDBC(o_hdbc2);

        new_order_rows_loaded++;

        CheckForCommit(o_hdbc2, o_hstmt2,
new_order_rows_loaded, "new_order",
&new_order_time_start->time_start);
    }

    // rcint = bcp_batch(o_hdbc2);
    // if (rcint < 0)
    //     HandleErrorDBC(o_hdbc2);

    if ((o_w_id == aptr->num_warehouses) &&
(o_d_id == 10))
    {
        rcint = bcp_done(o_hdbc2);

        if (rcint < 0)

            HandleErrorDBC(o_hdbc2);

        SQLFreeStmt(o_hstmt2, SQL_DROP);
        SQLDisconnect(o_hdbc2);
        SQLFreeConnect(o_hdbc2);

        // if build index after load...
        if ((aptr->build_index == 1) &&
(aptr->index_order == 0))
            BuildIndex("idxnodc1");
    }
}

//=====
//
// Function : LoadOrderLineTable
//
//=====

void LoadOrderLineTable(LOADER_TIME_STRUCT
*order_line_time_start)
{
    int i,j;
    long o_id;
    short o_d_id;
    short o_w_id;
    long ol;
    long ol_i_id;

```

```

short      ol_supply_w_id;
short      ol_quantity;
double     ol_amount;
char       ol_dist_info[DIST_INFO_LEN+1];
char       ol_delivery_d[OL_DELIVERY_D_LEN+1];
RETCODE    rc;
DBINT      rcint;

// bind ORDER-LINE data
rc = bcp_bind(o_hdbc3, (BYTE *) &o_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_d_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &o_w_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 4);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_i_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 5);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_supply_w_id,
0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 6);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *)
&ol_delivery_d, 0, OL_DELIVERY_D_LEN, NULL, 0,
SQLCHARACTER, 7);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_quantity, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT2, 8);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) &ol_amount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

rc = bcp_bind(o_hdbc3, (BYTE *) ol_dist_info, 0,
DIST_INFO_LEN, NULL, 0, 0, 10);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);

for (i = 0; i < orders_per_district; i++)

```

```

        o_id = orders_buf[i].o_id;
        o_d_id = orders_buf[i].o_d_id;
        o_w_id = orders_buf[i].o_w_id;

        for (j=0; j <
orders_buf[i].o_ol_cnt; j++)
        {
            ol =
orders_buf[i].o_ol[j].ol;
            ol_i_id =
orders_buf[i].o_ol[j].ol_i_id;
            ol_supply_w_id =
orders_buf[i].o_ol[j].ol_supply_w_id;
            ol_quantity =
orders_buf[i].o_ol[j].ol_quantity;
            ol_amount =
orders_buf[i].o_ol[j].ol_amount;

            strcpy(ol_delivery_d,orders_buf[i].o_ol[j].
ol_delivery_d);

            strcpy(ol_dist_info,orders_buf[i].o_ol[j].o
l_dist_info);

            rc =
bcp_sendrow(o_hdbc3);
            if (rc != SUCCEED)

                HandleErrorDBC(o_hdbc3);

            order_line_rows_loaded++;
            CheckForCommit(o_hdbc3,
o_hstmt3, order_line_rows_loaded, "order_line",
&order_line_time_start->time_start);
        }

        // rcint = bcp_batch(o_hdbc3);
        // if (rcint < 0)
        //     HandleErrorDBC(o_hdbc3);

        if ((o_w_id == aptr->num_warehouses) &&
(o_d_id == 10))
        {
            rcint = bcp_done(o_hdbc3);

            if (rcint < 0)

                HandleErrorDBC(o_hdbc3);

            SQLFreeStmt(o_hstmt3, SQL_DROP);
            SQLDisconnect(o_hdbc3);
            SQLFreeConnect(o_hdbc3);

            // if build index after load...
            if ((aptr->build_index == 1) &&
(aptr->index_order == 0))

```

```

                BuildIndex("idxodlcl");
            }
        }

//=====
//
// Function : GetPermutation
//
//=====
void GetPermutation(int perm[], int n)
{
    int i, r, t;

    for (i=1;i<=n;i++)
        perm[i] = i;

    for (i=1;i<=n;i++)
    {
        r = RandomNumber(i,n);
        t = perm[i];
        perm[i] = perm[r];
        perm[r] = t;
    }
}

//=====
//
// Function : CheckForCommit
//
//=====
void CheckForCommit(HDBC hdbc,

                    HSTMT hstmt,

                    int

                    rows_loaded,

                    char *table_name,

                    long

                    *time_start)
{
    long time_end, time_diff;
    // DBINT rcint;

    if ( !(rows_loaded % aptr->batch) )
    {
        // rcint = bcp_batch(hdbc);
        // if (rcint < 0)
        //     HandleErrorDBC(hdbc);

```

```

        time_end = (TimeNow() / MILLI);
        time_diff = time_end -
*time_start;

        printf("-> Loaded %ld rows into
%s in %ld sec - Total = %d (%.2f rps)\n",
        aptr->batch,
        table_name,
        time_diff,
        rows_loaded,
        (float) aptr->batch / (time_diff ? time_diff : 1L));

        *time_start = time_end;
    }
    return;
}

//=====
//
// Function : OpenConnections
//
//=====
void OpenConnections()
{
    RETCODE rc;

    char
    szDriverString[300];
    char
    szDriverStringOut[1024];
    SQLSMALLINT
    cbDriverStringOut;

    SQLAllocHandle(SQL_HANDLE_ENV,
SQL_NULL_HANDLE, &henv );

    SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION,
(void*)SQL_OV_ODBC3, 0 );

    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&i_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&w_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&c_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&c_hdbc2);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&o_hdbc1);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&o_hdbc2);
    SQLAllocHandle(SQL_HANDLE_DBC, henv ,
&o_hdbc3);

```

```

        SQLSetConnectAttr(i_hdbc1, SQL_COPT_SS_BCP,
(void *)SQL_BCP_ON, SQL_IS_INTEGER );
        SQLSetConnectAttr(w_hdbc1, SQL_COPT_SS_BCP,
(void *)SQL_BCP_ON, SQL_IS_INTEGER );
        SQLSetConnectAttr(c_hdbc1, SQL_COPT_SS_BCP,
(void *)SQL_BCP_ON, SQL_IS_INTEGER );
        SQLSetConnectAttr(c_hdbc2, SQL_COPT_SS_BCP,
(void *)SQL_BCP_ON, SQL_IS_INTEGER );
        SQLSetConnectAttr(o_hdbc1, SQL_COPT_SS_BCP,
(void *)SQL_BCP_ON, SQL_IS_INTEGER );
        SQLSetConnectAttr(o_hdbc2, SQL_COPT_SS_BCP,
(void *)SQL_BCP_ON, SQL_IS_INTEGER );
        SQLSetConnectAttr(o_hdbc3, SQL_COPT_SS_BCP,
(void *)SQL_BCP_ON, SQL_IS_INTEGER );

        // Open connections to SQL Server
        // Connection 1

        sprintf( szDriverString, "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,

        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );

        rc = SQLSetConnectOption (i_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(i_hdbc1);

        rc = SQLDriverConnect ( i_hdbc1,
NULL,

        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,

        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,

        SQL_DRIVER_NOPROMPT );
        if (rc != SUCCEEDED)
            HandleErrorDBC(i_hdbc1);

        // Connection 2

        sprintf( szDriverString, "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,

        aptr->server,
        aptr->user,
        aptr->password,

```

```

        aptr->database );

        rc = SQLSetConnectOption (w_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        rc = SQLDriverConnect ( w_hdbc1,
NULL,

        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,

        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,

        SQL_DRIVER_NOPROMPT );
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        // Connection 3

        sprintf( szDriverString, "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,

        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );

        rc = SQLSetConnectOption (c_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        rc = SQLDriverConnect ( c_hdbc1,
NULL,

        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,

        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,

        SQL_DRIVER_NOPROMPT );
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        // Connection 4

```

```

        sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );
rc = SQLSetConnectOption ( c_hdbc2,
SQL_PACKET_SIZE, aptr->pack_size);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc2);
rc = SQLDriverConnect ( c_hdbc2,
    NULL,
    (SQLCHAR*)&szDriverString[0] ,
    SQL_NTS,
    (SQLCHAR*)&szDriverStringOut[0],
    sizeof(szDriverStringOut),
    &cbDriverStringOut,
    SQL_DRIVER_NOPROMPT );
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc2);
// Connection 5
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );
rc = SQLSetConnectOption ( o_hdbc1,
SQL_PACKET_SIZE, aptr->pack_size);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc1);
rc = SQLDriverConnect ( o_hdbc1,
    NULL,
    (SQLCHAR*)&szDriverString[0] ,
    SQL_NTS,
    (SQLCHAR*)&szDriverStringOut[0],

```

```

    sizeof(szDriverStringOut),
    &cbDriverStringOut,
    SQL_DRIVER_NOPROMPT );
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc1);
// Connection 6
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );
rc = SQLSetConnectOption ( o_hdbc2,
SQL_PACKET_SIZE, aptr->pack_size);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc2);
rc = SQLDriverConnect ( o_hdbc2,
    NULL,
    (SQLCHAR*)&szDriverString[0] ,
    SQL_NTS,
    (SQLCHAR*)&szDriverStringOut[0],
    sizeof(szDriverStringOut),
    &cbDriverStringOut,
    SQL_DRIVER_NOPROMPT );
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc2);
// Connection 7
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );
rc = SQLSetConnectOption ( o_hdbc3,
SQL_PACKET_SIZE, aptr->pack_size);
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);

```

```

rc = SQLDriverConnect ( o_hdbc3,
    NULL,
    (SQLCHAR*)&szDriverString[0] ,
    SQL_NTS,
    (SQLCHAR*)&szDriverStringOut[0],
    sizeof(szDriverStringOut),
    &cbDriverStringOut,
    SQL_DRIVER_NOPROMPT );
if (rc != SUCCEEDED)
    HandleErrorDBC(o_hdbc3);
}
//=====
//
// Function name: BuildIndex
//
//=====
void BuildIndex(char          *index_script)
{
    char      cmd[256];
        printf("Starting index creation:
%s\n",index_script);
        sprintf(cmd, "isql -S%s -U%s -P%s -e -
i%s\\%s.sql > logs\\%s.log",
                                aptr->server,
                                aptr->user,
                                aptr->password,
                                aptr->index_script_path,
                                index_script,
                                index_script);
        system(cmd);
        printf("Finished index creation:
%s\n",index_script);
}
void HandleErrorDBC (SQLHDBC hdbc1)
{
    SQLCHAR      SqlState[6],
    Msg[SQL_MAX_MESSAGE_LENGTH];
    SQLINTEGER   NativeError;
    SQLSMALLINT  i, MsgLen;
    SQLRETURN    rc2;
    char         timebuf[128];

```

```

char          datebuf[128];
FILE          *fp1;

i = 1;
while (( rc2 = SQLGetDiagRec(SQL_HANDLE_DBC
, hdbc1, i, SqlState , &NativeError,
Msg,
sizeof(Msg) , &MsgLen ) != SQL_NO_DATA )
{
    sprintf( szLastError , "%s" ,
Msg );

    _strtime(timebuf);
    _strdate(datebuf);

    printf( "[%s : %s] %s\n" ,
datebuf, timebuf, szLastError);

    fp1 =
fopen("logs\\tpccldr.err","w");
    if (fp1 == NULL)
        printf("ERROR: Unable
to open errorlog file.\n");
    else
    {
        fprintf(fp1, "[%s : %s]
%s\n" , datebuf, timebuf, szLastError);
        fclose(fp1);
    }
    i++;
}

void FormatDate ( char* szTimeCOutput )
{
    struct tm when;
    time_t now;

    time( &now );
    when = *localtime( &now );

    mktime( &when );

    // odbc datetime format
    strftime( szTimeCOutput , 30 , "%Y-%m-%d
%H:%M:%S.000" , &when );

    return;
}

```

time.c

```

// File:          TIME.C
//
// TPC-C Kit Ver. 4.00          Microsoft

```

```

//
// Copyright
// Microsoft, 1996, 1997, 1998
// Purpose: Source file for time functions

// Includes
#include "tpcc.h"

// Globals
static long start_sec;

//=====
//
// Function name: TimeNow
//
//=====

long TimeNow()
{
    long          time_now;
    struct _timeb el_time;

#ifdef DEBUG
    printf("[%ld]DBG: Entering TimeNow()\n", (int)
GetCurrentThreadId());
#endif

    _ftime(&el_time);

    time_now = ((el_time.time - start_sec) * 1000) +
el_time.millitm;

    return time_now;
}

```

strings.c

```

// File:          STRINGS.C
//
// Microsoft
// TPC-C Kit Ver. 4.00
// Copyright
// Microsoft, 1996, 1997, 1998
// Purpose: Source file for database loader
// string functions

// Includes
#include "tpcc.h"
#include <string.h>
#include <ctype.h>

//=====
//
// Function name: MakeAddress
//

```

```

//=====
//
void MakeAddress(char *street_1,
char
*street_2,
char *city,
char *state,
char *zip)
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeAddress()\n",
(int) GetCurrentThreadId());
#endif

    MakeAlphaString(10, 20, ADDRESS_LEN, street_1);
    MakeAlphaString(10, 20, ADDRESS_LEN, street_2);
    MakeAlphaString(10, 20, ADDRESS_LEN, city);
    MakeAlphaString( 2, 2, STATE_LEN, state);
    MakeZipNumberString( 9, 9, ZIP_LEN, zip);

#ifdef DEBUG
    printf("[%ld]DBG: MakeAddress: street_1: %s,
street_2: %s, city: %s, state: %s, zip: %s\n",
(int)
GetCurrentThreadId(), street_1, street_2, city,
state, zip);
#endif

    return;
}

//=====
//
// Function name: LastName
//
//=====

void LastName(int num,
char *name)
{
    static char *n[] =
{
        "BAR" , "OUGHT" , "ABLE" , "PRI"
, "PRES",
        "ESE" , "ANTI" , "CALLY",
"ATION", "EING"
    };

#ifdef DEBUG
    printf("[%ld]DBG: Entering LastName()\n", (int)
GetCurrentThreadId());
#endif

    if ((num >= 0) && (num < 1000))
    {
        strcpy(name, n[(num/100)%10]);
        strcat(name, n[(num/10)%10]);
    }
}

```

```

        strcat(name, n[(num/1)%10]);
    if (strlen(name) < LAST_NAME_LEN)
    {
        PaddString(LAST_NAME_LEN, name);
    }
    else
    {
        printf("\nError in LastName()...
num <%ld> out of range (0,999)\n", num);
        exit(-1);
    }

#ifdef DEBUG
    printf("[%ld]DBG: LastName: num = [%d] ==>
[%d][%d][%d]\n",
            (int)
GetCurrentThreadId(), num, num/100, (num/10)%10,
num%10);
    printf("[%ld]DBG: LastName: String = %s\n",
(int) GetCurrentThreadId(), name);
#endif

    return;
}

//=====
//
// Function name: MakeAlphaString
//
//=====
//philipdu 08/13/96 Changed MakeAlphaString to use A-
Z, a-z, and 0-9 in
//accordance with spec see below:
//The spec says:
//4.3.2.2 The notation random a-string [x .. y]
//(respectively, n-string [x .. y]) represents a
string of random alphanumeric
//(respectively, numeric) characters of a random
length of minimum x, maximum y,
//and mean (y+x)/2. Alphanumerics are A..Z, a..z, and
0..9. The only other
//requirement is that the character set used "must be
able to represent a minimum
//of 128 different characters". We are using 8-bit
chars, so this is a non issue.
//It is completely unreasonable to stuff non-printing
chars into the text fields.
//-CLevine 08/13/96

int MakeAlphaString( int x, int y, int z, char
*str)
{
    int len;

```

```

        int i;
        static char chArray[] =
"0123456789ABCDEFGHIJKLMNPOQRSTUVWXYZabcdefghijklmnop
qrstuvwxyz";
        static int chArrayMax = 61;

#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeAlphaString()\n",
(int) GetCurrentThreadId());
#endif

    len= RandomNumber(x, y);

    for (i=0; i<len; i++)
        str[i] = chArray[RandomNumber(0,
chArrayMax)];
    if ( len < z )
        memset(str+len, ' ', z - len);
    str[len] = 0;

    return len;
}

//=====
//
// Function name: MakeOriginalAlphaString
//
//=====
int MakeOriginalAlphaString(int x,
        int y,
        int z,
        char *str,
        int percent)
{
    int len;
    int val;
    int start;

#ifdef DEBUG
    printf("[%ld]DBG: Entering
MakeOriginalAlphaString()\n", (int)
GetCurrentThreadId());
#endif

    // verify percentage is valid
    if ((percent < 0) || (percent > 100))
    {
        printf("MakeOriginalAlphaString:
Invalid percentage: %d\n", percent);
        exit(-1);
    }

    // verify string is at least 8 chars in length
    if ((x + y) <= 8)
    {

```

```

        printf("MakeOriginalAlphaString:
string length must be >= 8\n");
        exit(-1);
    }

    // Make Alpha String
    len = MakeAlphaString(x,y, z, str);

    val = RandomNumber(1,100);
    if (val <= percent)
    {
        start = RandomNumber(0, len - 8);
        strncpy(str + start, "ORIGINAL",
8);
    }

#ifdef DEBUG
    printf("[%ld]DBG: MakeOriginalAlphaString: :
%s\n",
            (int)
GetCurrentThreadId(), str);
#endif

    return strlen(str);
}

//=====
//
// Function name: MakeNumberString
//
//=====
int MakeNumberString(int x, int y, int z, char
*str)
{
    char tmp[16];

    //MakeNumberString is always called
MakeZipNumberString(16, 16, 16, string)

    memset(str, '0', 16);
    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str+8, tmp, strlen(tmp));

    str[16] = 0;

    return 16;
}

//=====
//
// Function name: MakeZipNumberString
//
//=====

```



```

int MakeZipNumberString(int x, int y, int z, char
*str)
{
    char tmp[16];

    //MakeZipNumberString is always called
MakeZipNumberString(9, 9, 9, string)

    strcpy(str, "000011111");

    itoa(RandomNumber(0, 9999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

    return 9;
}

//=====
//
// Function name: InitString
//
//=====
void InitString(char *str, int len)
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering InitString()\n", (int)
GetCurrentThreadId());
#endif

    memset(str, ' ', len);
    str[len] = 0;
}

//=====
//
// Function name: InitAddress
//
// Description:
//
//=====
void InitAddress(char *street_1, char *street_2, char
*city, char *state, char *zip)
{
    memset(street_1, ' ', ADDRESS_LEN+1);
    memset(street_2, ' ', ADDRESS_LEN+1);
    memset(city, ' ', ADDRESS_LEN+1);

    street_1[ADDRESS_LEN+1] = 0;
    street_2[ADDRESS_LEN+1] = 0;
    city[ADDRESS_LEN+1] = 0;

    memset(state, ' ', STATE_LEN+1);
    state[STATE_LEN+1] = 0;

    memset(zip, ' ', ZIP_LEN+1);
    zip[ZIP_LEN+1] = 0;
}

```

```

//=====
//
// Function name: PaddString
//
//=====
void PaddString(int max, char *name)
{
    int len;

    len = strlen(name);
    if ( len < max )
        memset(name+len, ' ', max - len);
    name[max] = 0;

    return;
}

random.c

// File: RANDOM.C Microsoft
// TPC-C Kit Ver. 4.00 Copyright
// Microsoft, 1996, 1997, 1998
// Purpose: Random number generation routines
for database loader

// Includes
#include "tpcc.h"
#include "math.h"

// Defines
#define A 16807
#define M 2147483647
#define Q 127773 /* M div A */
#define R 2836 /* M mod A */
#define Thread __declspec(thread)

// Globals
long Thread Seed = 0; /* thread local seed */

/*****
*****
*
* random -
*
* Implements a GOOD pseudo random number
generator. This generator
* will/should? run the complete period before
repeating.
*
* Copied from:
*

```

```

* Random Numbers Generators: Good Ones Are Hard
to Find.
* Communications of the ACM - October 1988
Volume 31 Number 10
*
* Machine Dependencies:
*
* long must be 2 ^ 31 - 1 or greater.
*
*
*
*****/

/*****
*****
* seed - load the Seed value used in irand and drand.
Should be used before
* first call to irand or drand.
*
*****/

void seed(long val)
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering seed()...\n", (int)
GetCurrentThreadId());
    printf("Old Seed %ld New Seed %ld\n",Seed,
val);
#endif

    if ( val < 0 )
        val = abs(val);

    Seed = val;
}

/*****
*****
*
* irand - returns a 32 bit integer pseudo random
number with a period of
* 1 to 2 ^ 32 - 1.
*
*
* parameters:
*
* none.
*
*
* returns:
*
* 32 bit integer - defined as long ( see above
).
*

```

```

*
*
* side effects:
*
*      seed get recomputed.
*
*****
*****/

long irand()
{
    register long  s;    /* copy of seed */
    register long  test; /* test flag */
    register long  hi;   /* tmp value for speed */
    register long  lo;   /* tmp value for speed */

#ifdef DEBUG
    printf("[%ld]DBG: Entering irand()...\n", (int)
GetCurrentThreadId());
#endif

    s = Seed;
    hi = s / Q;
    lo = s % Q;

    test = A * lo - R * hi;
    if ( test > 0 )
        Seed = test;
    else
        Seed = test + M;

    return( Seed );
}

/*****
*****
*
* drand - returns a double pseudo random number
between 0.0 and 1.0.
*      See irand.
*
*****
*****/

double drand()
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering drand()...\n", (int)
GetCurrentThreadId());
#endif

    return( (double)irand() / 2147483647.0);
}

//=====
// Function : RandomNumber
//

```

```

// Description:
//=====
long RandomNumber(long lower, long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering RandomNumber()...\n",
(int) GetCurrentThreadId());
#endif

    if ( upper == lower ) /* pgd 08-13-
96 perf enhancement */
        return lower;

    upper++;

    if ( upper <= lower )
        rand_num = upper;
    else
        rand_num = lower + irand() %
(upper - lower); /* pgd 08-13-96 perf enhancement */

#ifdef DEBUG
    printf("[%ld]DBG: RandomNumber between %ld & %ld
==> %ld\n",
(int)
GetCurrentThreadId(), lower, upper, rand_num);
#endif

    return rand_num;
}

#if 0
//Original code pgd 08/13/96

long RandomNumber(long lower,
long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering RandomNumber()...\n",
(int) GetCurrentThreadId());
#endif

    upper++;

    if ((upper <= lower))
        rand_num = upper;
    else
        rand_num = lower + irand() %
((upper > lower) ? upper - lower : upper);

#ifdef DEBUG
    printf("[%ld]DBG: RandomNumber between %ld & %ld
==> %ld\n",

```

```

(int)
GetCurrentThreadId(), lower, upper, rand_num);
#endif

    return rand_num;
}
#endif

//=====
// Function : NURand
//
// Description:
//=====
long NURand(int iConst,
long x,
long y,
long C)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering NURand()...\n", (int)
GetCurrentThreadId());
#endif

    rand_num = (((RandomNumber(0,iConst) |
RandomNumber(x,y) + C) % (y-x+1))+x);

#ifdef DEBUG
    printf("[%ld]DBG: NURand: num = %d\n", (int)
GetCurrentThreadId(), rand_num);
#endif

    return rand_num;
}

-----
getargs.c
-----
// File: GETARGS.C Microsoft
// TPC-C Kit Ver. 4.00 Copyright
// Microsoft, 1996, 1997, 1998
// Purpose: Source file for command line
processing

// Includes
#include "tpcc.h"

//=====
//
// Function name: GetArgsLoader
//
//=====

```

```

void GetArgsLoader(int argc, char **argv,
TPCC_LDR_ARGS *pargs)
{
    int i;
    char *ptr;

#ifdef DEBUG
    printf("[%ld]DBG: Entering GetArgsLoader()\n",
(int) GetCurrentThreadId());
#endif

    /* init args struct with some useful values */
    pargs->server = SERVER;
    pargs->user = USER;
    pargs->password = PASSWORD;
    pargs->database = DATABASE;
    pargs->batch = BATCH;
    pargs->num_warehouses = UNDEF;
    pargs->tables_all =
TRUE;
    pargs->table_item =
FALSE;
    pargs->table_warehouse =
FALSE;
    pargs->table_customer =
FALSE;
    pargs->table_orders =
FALSE;
    pargs->loader_res_file =
LOADER_RES_FILE;
    pargs->pack_size =
DEFLDPACKSIZE;
    pargs->starting_warehouse =
DEF_STARTING_WAREHOUSE;
    pargs->build_index =
BUILD_INDEX;
    pargs->index_order =
INDEX_ORDER;
    pargs->index_script_path =
INDEX_SCRIPT_PATH;
    pargs->scale_down =
SCALE_DOWN;

    /* check for zero command line args */
    if ( argc == 1 )
        GetArgsLoaderUsage();

    for ( i = 1; i < argc; ++i)
    {
        if (argv[i][0] != '-' &&
argv[i][0] != '/')
        {
            printf("\nUnrecognized command");
            GetArgsLoaderUsage();
            exit(1);
        }

        ptr = argv[i];

        switch (ptr[1])
        {
            case 'h': /* Fall through */

```

```

                case 'H':
                    GetArgsLoaderUsage();
                    break;

                case 'D':
                    pargs->
>database = ptr+2;
                    break;

                case 'P':
                    pargs->
>password = ptr+2;
                    break;

                case 'S':
                    pargs->server
= ptr+2;
                    break;

                case 'U':
                    pargs->user =
ptr+2;
                    break;

                case 'b':
                    pargs->batch
= atol(ptr+2);
                    break;

                case 'W':
                    pargs->
>num_warehouses = atol(ptr+2);
                    break;

                case 's':
                    pargs->
>starting_warehouse = atol(ptr+2);
                    break;

                case 't':
                    {
                        pargs->tables_all = FALSE;
                        if
(strncmp(ptr+2,"item") == 0)
                            pargs->table_item = TRUE;
                        else if (strncmp(ptr+2,"warehouse") == 0)
                            pargs->table_warehouse = TRUE;
                        else if (strncmp(ptr+2,"customer") == 0)
                            pargs->table_customer = TRUE;
                        else if (strncmp(ptr+2,"orders") == 0)
                            pargs->table_orders = TRUE;
                        else

```

```

                            printf("\nUnrecognized command");
                            GetArgsLoaderUsage();
                            exit(1);
                    }

                case 'f':
                    pargs->
>loader_res_file = ptr+2;
                    break;

                case 'p':
                    pargs->
>pack_size = atol(ptr+2);
                    break;

                case 'i':
                    pargs->
>build_index = atol(ptr+2);
                    break;

                case 'o':
                    pargs->
>index_order = atol(ptr+2);
                    break;

                case 'c':
                    pargs->
>scale_down = atol(ptr+2);
                    break;

                case 'd':
                    pargs->
>index_script_path = ptr+2;
                    break;

                default:
                    GetArgsLoaderUsage();
                    exit(-1);
                    break;
            }

        /* check for required args */
        if (pargs->num_warehouses == UNDEF )
        {
            printf("Number of Warehouses is
required\n");
            exit(-2);
        }

        return;
    }

    //=====
    =====

```

```

//
// Function name: GetArgsLoaderUsage
//
//=====
void GetArgsLoaderUsage()
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering
GetArgsLoaderUsage()\n", (int) GetCurrentThreadId());
#endif

    printf("TPCCLDR:\n\n");
    printf("Parameter
Default\n");
    printf("-----\n");
    printf("-W Number of Warehouses to Load
Required \n");
    printf("-S Server
%s\n", SERVER);
    printf("-U Username
%s\n", USER);
    printf("-P Password
%s\n", PASSWORD);
    printf("-D Database
%s\n", DATABASE);
    printf("-b Batch Size
%ld\n", (long) BATCH);
    printf("-p TDS packet size
%ld\n", (long) DEFLDPPACKSIZE);
    printf("-f Loader Results Output Filename
%s\n", LOADER_RES_FILE);
    printf("-s Starting Warehouse
%ld\n", (long) DEF_STARTING_WAREHOUSE);
    printf("-i Build Option (data = 0, data and
index = 1) %ld\n", (long) BUILD_INDEX);
    printf("-o Cluster Index Build Order
(before = 1, after = 0) %ld\n", (long) INDEX_ORDER);
    printf("-c Build Scaled Database (normal =
0, tiny = 1) %ld\n", (long) SCALE_DOWN);
    printf("-d Index Script Path
%s\n", INDEX_SCRIPT_PATH);
    printf("-t Table to Load
all tables \n");
    printf("    [item|warehouse|customer|orders]\n");
    printf("    Notes: \n");
    printf("    - the '-t' parameter may be included
multiple times to \n");
    printf("    specify multiple tables to be
loaded \n");
    printf("    - 'item' loads ITEM table \n");
    printf("    - 'warehouse' loads WAREHOUSE,
DISTRICT, and STOCK tables \n");
    printf("    - 'customer' loads CUSTOMER and
HISTORY tables \n");
    printf("    - 'orders' load NEW-ORDER, ORDERS,
ORDER-LINE tables \n");

```

```

        printf("\nNote: Command line switches are
case sensitive.\n");
    }
    exit(0);
}

```

Appendix C: *Tunable Parameters*

Microsoft Windows NT v. 4.0 Configuration Parameters

The default value for the ProcessorAffinityMask was changed as shown below (using the regedt32 command)

```
[HKEY_LOCAL_MACHINE\SYSTEM\
CurrentControlSet\Services\NDIS\Parameters]
"ProcessorAffinityMask"=dword:00000000
```

The details about this parameter is fully documented in Microsoft Windows NT Server resource Kit (CD) - Online Help File (REGENRY.HLP)

Microsoft SQL Server Startup Parameters

```
C:\MSSQL7\BINN\SQLSERVER
-eC:\MSSQL7\LOG\ERRORLOG -x -c -t3502
-g40
```

Where:

- -c Start SQL Server independently of the Windows NT Service Control Manager
- -x Disables the keeping of CPU time and cache-hit ratio statistics

- -t3502 Prints a message to the SQL Server log at the start and end of each checkpoint
- -g40 Specifies the amount of memory in megabytes to reserve for memory requests outside of the buffer pool.

Microsoft Windows NT v. 4.0 Configuration Parameters

The following services were disabled in the Windows NT Control Panel/Services on both the Server and the Client systems:

- Computer Browser
- License Logging Service
- Messenger
- NT LM Security Support Provider
- Remote Procedure Call (RPC) Service
- Server
- Spooler
- TPC/IP Netbios Helper

SQL Server Stack Size

The default stack size for Microsoft SQL Server was changed using the EDITBIN utility. The EDITBIN utility ships with Microsoft Visual C++ V4.0. The command used to change the stack size is:

```
editbin /Stack:131072 sqlservr.exe
```

This command is fully documented as an article in the Microsoft Knowledge Base on the Microsoft Web Site at www.microsoft.com/support.

Compaq Device Drivers for Windows NT

The following Microsoft Windows NT device drivers were replaced with Compaq-specific device drivers:

- The Microsoft SMART-3200 Array Controller default device driver (CPQARRAY.SYS) was replaced with the Compaq SMART-3200 Array Controller ECI Driver version 1.06 for Microsoft Windows NT 4.0 (CPQSMRT2.SYS). This driver is available on the Compaq Computer Corporation World Wide Web site (www.compaq.com). It is distributed via Softpaq SP7561

Client Configuration

The configurations in the following sections were captured from one client machine and apply to all four clients: Internet Information Server Registry Parameters, World Wide Web Service Registry Parameters, TPCC Application Registry Parameters, Tuxedo UBB File, Client Systems Configuration.

Microsoft SQL Server 7.0 Configuration Parameters

```

1> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11>
-- File:      VERSION.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Returns SQL Server version string

```

```

print " "
select convert(char(30), getdate(),9)
print " "

```

```

-----
Feb 2 1999 2:53:33:483PM

```

```

(1 row affected)

```

```

1> 2> 3>
select @@version

```

```

-----
-----
-----
Microsoft SQL Server 7.00 - 7.00.623 (Intel X86)
Nov 13 1998 02:37:14
Cop
yright (c) 1988-1998 Microsoft Corporation
Enterprise Edition on Windo
ws NT 4.0 (Build 1381: Service Pack 4)

(1 row affected)
1> 2>
1> 2> 3> 4> 5> 6> 7> 8> 9> 10>
-- File:      CONFIG.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Collects SQL Server configuration
parameters

print " "
select convert(char(30), getdate(),9)
print " "

-----
Feb 2 1999 2:53:34:280PM

(1 row affected)

1> 2> 3> DBCC execution completed. If DBCC printed
error messages, contact your system administrator.
Configuration option changed. Run the RECONFIGURE
statement to install.

sp_configure "show advanced",1
1> 2> reconfigure with override
1> 2> sp_configure
name
minimum      maximum      config_value run_value
-----
-----
-----
affinity mask
0 2147483647      15      15
allow updates
0 1 0 0
cost threshold for parallelism
0 32767 5 5
cursor threshold
-1 2147483647 -1 -1
default language
0 9999 0 0
default sortorder id
0 255 50 50
extended memory size (MB)
0 2147483647 0 0

```

```

fill factor (%)
0 100 0 0
index create memory (KB)
704 1600000 704 704
language in cache
3 100 3 3
language neutral full-text
0 1 0 0
lightweight pooling
0 1 1 1
locks
5000 2147483647 15000 15000
max async IO
1 255 255 255
max degree of parallelism
0 32 1 1
max server memory (MB)
4 2147483647 3000 3000
max text repl size (B)
0 2147483647 65536 65536
max worker threads
10 1024 200 200
media retention
0 365 0 0
min memory per query (KB)
512 2147483647 512 512
min server memory (MB)
0 2147483647 3000 3000
nested triggers
0 1 1 1
network packet size (B)
512 65535 4096 4096
open objects
0 2147483647 0 0
priority boost
0 1 1 1
query governor cost limit
0 2147483647 0 0
query wait (s)
-1 2147483647 -1 -1
recovery interval (min)
0 32767 32767 32767
remote access
0 1 1 1
remote login timeout (s)
0 2147483647 5 5
remote proc trans
0 1 0 0
remote query timeout (s)
0 2147483647 0 0
resource timeout (s)
5 2147483647 10 10
scan for startup procs
0 1 0 0
set working set size
0 1 1 1
show advanced options
0 1 1 1
spin counter
1 2147483647 10000 10000
time slice (ms)
50 1000 200 200

```

```

two digit year cutoff
1753      9999      2049      2049
Unicode comparison style
0 2147483647      0      0
Unicode locale id
0 2147483647      33280      33280
user connections
0 32767      0      0
user options
0 4095      0      0

```

Internet Information Server Registry Parameters

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]
"BandwidthLevel"=dword:ffffffff
"ListenBackLog"=dword:00001770
"PoolThreadLimit"=dword:00000a28
"ThreadTimeout"=dword:00030d40
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\Filter]
"FilterType"=dword:00000000
"NumGrantSites"=dword:00000000
"NumDenySites"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters\MimeMap]
"text/html,htm,,h"=""
"image/gif,gif,,g"=""
"image/jpeg,jpg,,:"=""
"text/plain,txt,,0"=""
"text/html,html,,h"=""
"image/jpeg,jpeg,,:"=""
"image/jpeg,jpe,,:"=""
"image/bmp,bmp,,:"=""
"application/octet-stream,*,,5"=""
"application/pdf,pdf,,5"=""
"application/octet-stream,bin,,5"=""
"application/oda,oda,,5"=""
"application/zip,zip,,9"=""
"application/rtf,rtf,,5"=""
"application/postscript,ps,,5"=""
"application/postscript,ai,,5"=""
"application/postscript,eps,,5"=""
"application/mac-binhex40,hqx,,4"=""
"application/msword,doc,,5"=""
```

```

"application/msword,dot,,5"=""
"application/winhelp,hlp,,5"=""
"video/mpeg,mpeg,,:"=""
"video/mpeg,mpg,,:"=""
"video/mpeg,mpe,,:"=""
"video/x-msvideo,avi,,<"=""
"video/quicktime,qt,,:"=""
"video/quicktime,mov,,:"=""
"video/x-sgi-movie,movie,,<"=""
"x-world/x-vrml,wrl,,5"=""
"x-world/x-vrml,xaf,,5"=""
"x-world/x-vrml,xof,,5"=""
"x-world/x-vrml,flr,,5"=""
"x-world/x-vrml,wrz,,5"=""
"application/x-director,dcr,,5"=""
"application/x-director,dir,,5"=""
"application/x-director,dxr,,5"=""
"image/cis-cod,cod,,5"=""
"image/x-cmx,cmx,,5"=""
"application/envoy,evy,,5"=""
"application/msaccess,mdb,,5"=""
"application/x-mscardfile,crd,,5"=""
"application/x-msclip,clip,,5"=""
"application/octet-stream,exe,,5"=""
"application/x-msexcel,xla,,5"=""
"application/x-msexcel,xlc,,5"=""
"application/x-msexcel,xlm,,5"=""
"application/x-msexcel,xls,,5"=""
"application/x-msexcel,xlt,,5"=""
"application/x-msexcel,xlw,,5"=""
"application/x-msmediaview,m13,,5"=""
"application/x-msmediaview,m14,,5"=""
"application/x-msmoney,mny,,5"=""
"application/x-mspowerpoint,ppt,,5"=""
"application/x-msproject,mpp,,5"=""
"application/x-mspublisher,pub,,5"=""
"application/x-sterminal,term,,5"=""
"application/x-msworks,wks,,5"=""
"application/x-mswrite,wri,,5"=""
"application/x-msmetafile,wmf,,5"=""
"application/x-csh,csh,,5"=""
"application/x-dvi,dvi,,5"=""
"application/x-hdf,hdf,,5"=""
"application/x-latex,latex,,5"=""
"application/x-netcdf,nc,,5"=""
"application/x-netcdf,cdf,,5"=""
"application/x-sh,sh,,5"=""
"application/x-tcl,tcl,,5"=""
"application/x-tex,tex,,5"=""
"application/x-texinfo,texinfo,,5"=""
"application/x-texinfo,texi,,5"=""
"application/x-troff,t,,5"=""
"application/x-troff,tr,,5"=""
"application/x-troff,roff,,5"=""
"application/x-troff-man,man,,5"=""
"application/x-troff-me,me,,5"=""
"application/x-troff-ms,ms,,5"=""
"application/x-wais-source,src,,7"=""
"application/x-bcpio,bcpio,,5"=""
"application/x-cpio,cpio,,5"=""
"application/x-gtar,gtar,,9"=""
"application/x-shar,shar,,5"=""
"application/x-sv4cpio,sv4cpio,,5"=""

```

```

"application/x-sv4crc,sv4crc,,5"=""
"application/x-tar,tar,,5"=""
"application/x-ustar,ustar,,5"=""
"audio/basic,au,,<"=""
"audio/basic,snd,,<"=""
"audio/x-aiff,aiff,,<"=""
"audio/x-aiff,aifc,,<"=""
"audio/x-wav,wav,,<"=""
"audio/x-pn-realaudio,ram,,<"=""
"image/ief,ief,,:"=""
"image/tiff,tiff,,:"=""
"image/tiff,tif,,:"=""
"image/x-cmu-raster,ras,,:"=""
"image/x-portable-anymap,pnm,,:"=""
"image/x-portable-bitmap,pbm,,:"=""
"image/x-portable-graymap,pgm,,:"=""
"image/x-portable-pixmap,ppm,,:"=""
"image/x-rgb,rgb,,:"=""
"image/x-xbitmap,xbm,,:"=""
"image/x-xpixmap,xpm,,:"=""
"image/x-xwindowdump,xwd,,:"=""
"text/html,stm,,h"=""
"text/plain,bas,,0"=""
"text/plain,c,,0"=""
"text/plain,h,,0"=""
"text/richtext,rtx,,0"=""
"text/tab-separated-values,tsv,,0"=""
"text/x-setext,etx,,0"=""
"application/x-perfmon,pmc,,5"=""
"application/x-perfmon,pma,,5"=""
"application/x-perfmon,pmr,,5"=""
"application/x-perfmon,pml,,5"=""
"application/x-perfmon,pmw,,5"=""

```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Performance]
"Library"="infectrs.DLL"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"Last Counter"=dword:0000086c
"Last Help"=dword:0000086d
"First Counter"=dword:0000084e
"First Help"=dword:0000084f
```

World Wide Web Service Registry Parameters

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
```

```
"Start"=dword:00000002
"ErrorControl"=dword:00000000
"ImagePath"=hex(2):43,3a,5c,57,49,4e,4e,54,34,30,5c,5,
3,79,73,74,65,6d,33,32,5c,\
69,6e,65,74,73,72,76,5c,69,6e,65,74,69,6e,66,6f,2e,65,
78,65,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):52,50,43,53,53,00,4e,54,4c,4
d,53,53,50,00,00
"DependOnGroup"=hex(7):00
"ObjectName"="LocalSystem"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
\W3SVC\Parameters]
"MajorVersion"=dword:00000002
"MinorVersion"=dword:00000000
"AdminName"="Administrator"
"AdminEmail"="Admin@corp.com"
"MaxConnections"=dword:000186a0
"LogType"=dword:00000000
"LogFileDirectory"=hex(2):43,3a,5c,49,6e,65,74,4c,6f,
67,00
"LogFileTruncateSize"=dword:ffffffff
"LogFilePeriod"=dword:00000001
"LogFileFormat"=dword:00000000
"LogSqlDataSource"="HTTPLOG"
"LogSqlTableName"="Internetlog"
"LogSqlUserName"="InternetAdmin"
"LogSqlPassword"="sqllog"
"Authorization"=dword:00000001
"AnonymousUserName"="IUSR_CLIENT1"
"Default Load File"="Default.htm"
"Dir Browse Control"=dword:4000001e
"CheckForWAISDB"=dword:00000000
"CacheExtensions"=dword:00000001
"GlobalExpire"=dword:ffffffff
"ServerSideIncludesEnabled"=dword:00000001
"ServerSideIncludesExtension"=".stm"
"DebugFlags"=dword:00000008
"ScriptTimeout"=dword:00000384
"ConnectionTimeout"=dword:00000384
"InstallPath"="C:\\WINNT40\\System32\\inetsrv"
"SecurePort"=dword:000001bb
"Filter
DLLs"="C:\\WINNT40\\System32\\inetsrv\\sspicfilt.dll"
"AccessDeniedMessage"="Error: Access is Denied."
"NTAuthenticationProviders"="NTLM"
"ServerComment"=""
"AcceptExOutstanding"=dword:00001770
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
\W3SVC\Parameters\Script Map]
".idc"="C:\\WINNT40\\System32\\inetsrv\\httpodbc.dll"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
\W3SVC\Parameters\Virtual Roots]
"/,"="C:\\InetPub\\wwwroot,,5"
"/Scripts,"="C:\\InetPub\\scripts,,4"
"/iisadmin,"="C:\\WINNT40\\System32\\inetsrv\\iisadmi
n,,1"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
\W3SVC\Performance]
"Library"="W3ctrs.DLL"
"Open"="OpenW3PerformanceData"
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:000008a6
"Last Help"=dword:000008a7
"First Counter"=dword:0000086e
"First Help"=dword:0000086f
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
\W3SVC\Security]
"Security"=hex:01,00,14,80,c0,00,00,00,cc,00,00,00,14,
00,00,00,34,00,00,00,02,\
```

```
00,20,00,01,00,00,00,02,80,18,00,ff,01,0f,00,01,01,00,
00,00,00,00,01,00,00,\
```

```
00,00,20,02,00,00,02,00,8c,00,05,00,00,00,00,18,00,
8d,01,02,00,01,01,00,\
```

```
00,00,00,00,01,00,00,00,00,00,00,00,00,00,00,1c,00,fd,
01,02,00,01,02,00,00,\
```

```
00,00,00,05,20,00,00,00,23,02,00,00,d8,00,14,00,00,00,
1c,00,ff,01,0f,00,01,\
```

```
02,00,00,00,00,00,05,20,00,00,00,20,02,00,00,d8,00,14,
00,00,00,1c,00,ff,01,\
```

```
0f,00,01,02,00,00,00,00,05,20,00,00,00,25,02,00,00,
d8,00,14,00,00,00,18,\
```

```
00,fd,01,02,00,01,01,00,00,00,00,05,12,00,00,00,25,
02,00,00,01,01,00,00,\
```

```
00,00,00,05,12,00,00,00,01,01,00,00,00,00,05,12,00,
00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
\W3SVC\W3SAMP]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
\W3SVC\Enum]
"0"="Root\\LEGACY_W3SVC\\00000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

TPCC Application Registry Parameters

REGEDIT4

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\TPCC]
"PATH"="C:\\Inetpub\\wwwroot\\"
"LOG"="OFF"
"NumberOfDeliveryThreads"="7"
"MaximumWarehouses"="3000"
"BackoffDelay"="500"
"DeadlockRetry"="3"
"MaxConnections"="8000"
"LastInstalledVersion"="MTS"
"ConnectionPooling"="OFF"
"ConnectionPoolRetryTime"="500"
"QueueSlots"="330"
```

Tuxedo UBB File

```
*RESOURCES
IPCKEY 133133
```

```
MAXACCESSERS 2700
MAXSERVERS 180
MAXSERVICES 1800
MODEL SHM
MASTER C2
LDBAL Y
SCANUNIT 60
BLOCKTIME 60
BBLQUERY 60
```

```
*MACHINES
DEFAULT:
```

```
"C2" LMID=C2
TUXDIR="C:\tuxedo"
APPDIR="C:\InetPub\wwwroot"
TUXCONFIG="C:\InetPub\wwwroot\tuxconfig"
ULOGPEX="C:\InetPub\wwwroot\ULOG"
TYPE="WinNT"
UID= 0
GID= 0
```

```
*GROUPS
```

```
Q1
LMID=C2 GRPNO=1 OPENINFO=NONE

Q2
LMID=C2 GRPNO=2 OPENINFO=NONE

Q3
LMID=C2 GRPNO=3 OPENINFO=NONE

Q4
LMID=C2 GRPNO=4 OPENINFO=NONE

Q5
LMID=C2 GRPNO=5 OPENINFO=NONE

Q6
LMID=C2 GRPNO=6 OPENINFO=NONE
```



```

Q7      LMID=C2 GRPNO=7 OPENINFO=NONE

Q8      LMID=C2 GRPNO=8 OPENINFO=NONE

Q9      LMID=C2 GRPNO=9 OPENINFO=NONE

Q10     LMID=C2 GRPNO=10 OPENINFO=NONE

GROUPDEL
LMID=C2 GRPNO=11 OPENINFO=NONE

*SERVERS
DEFAULT:

Q      SRVGRP=Q1
      SRVID=60
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q1q REPLYQ=Y

Q      SRVGRP=Q2
      SRVID=200
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q2q REPLYQ=Y

Q      SRVGRP=Q3
      SRVID=300
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q3q REPLYQ=Y

Q      SRVGRP=Q4
      SRVID=400
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q4q REPLYQ=Y

Q      SRVGRP=Q5
      SRVID=500
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q5q REPLYQ=Y

Q      SRVGRP=Q6
      SRVID=600
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q6q REPLYQ=Y

Q      SRVGRP=Q7
      SRVID=700
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q7q REPLYQ=Y

Q      SRVGRP=Q8
      SRVID=800

```

```

      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q8q REPLYQ=Y

Q      SRVGRP=Q9
      SRVID=900
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q9q REPLYQ=Y

Q      SRVGRP=Q10
      SRVID=1000
      MIN=4 MAX=8
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=q10q REPLYQ=Y

delivery SRVGRP=GROUPDEL
      SRVID=1100
      MIN=4 MAX=10
      CLOPT="-A -- -Sjavelina_ip -DTPCC"
      RQADDR=delq REPLYQ=N

```

*SERVICES

Client Systems Configuration

ProLiant 1600 is a trademark of Compaq Computer Corporation.

```

Date . . . . . 02/03/1999
Time . . . . . 18:12:57

Product . . . . . ProLiant 1600

Machine ID
  From System Board . . . . . CPQ0689

Processor . . . . . Pentium II(R) at
400 MHz
  Slot . . . . . 2
  Secondary Cache . . . . . 512K
  CPU ID . . . . . 0651

Processor . . . . . Pentium II(R) at
400 MHz
  Slot . . . . . 1
  Secondary Cache . . . . . 512K
  CPU ID . . . . . 0651

Processor(s) Mapped Out . . . . . None

Numeric Coprocessor . . . . . Integrated 387-
Compatible

Expansion Bus . . . . . ISA, PCI

System Identification Number . . . . . D826BWW10060

```

```

CPU Mode . . . . . Real Mode

System ROM
  Revision . . . . . 08/17/1998
  Family . . . . . P8
  Flashable . . . . . Yes
  Supports F10 partition . . . . . Yes

Video Controller ROM
  Revision . . . . . 1.31 (Cirrus)

Keyboard Controller ROM
  Revision . . . . . C.9 08/14/1996
  Family . . . . . K

Option ROMs
  Address Range . . . . . C0000 - C7FFF
  Data Dump . . . . . (CL-GD5446 PCI VGA
BIOS Version 1.31 Copyright 1992-1996...)

  Address Range . . . . . E8000 - EDFFF
  Data Dump . . . . . (Ver 1.28
(C) Copyright COMPAQ Computer Corporation 1998-All
R...)

Bootblock ROM . . . . . 04/07/1998

Standby Recovery Server
  Status . . . . . Disabled
  COM Port . . . . . COM1
  Server Configuration . . . . . Recovery
  Timeout Value . . . . . 1 minutes

Memory Boards Identified:
System Board
  DIMM Slot 1 (SDRAM) . . . . . 256 Megabytes
  DIMM Slot 2 (SDRAM) . . . . . 64 Megabytes
  DIMM Slot 3 (SDRAM) . . . . . 64 Megabytes
  DIMM Slot 4 . . . . . 0 Megabytes
Total Compaq Memory . . . . . 384 Megabytes

Keyboard . . . . . Enhanced

LPT Ports . . . . . LPT1 (Address 3BC)

COM Ports . . . . . COM1 (Address 3F8)
  COM2 (Address 2F8)

Diskette Drive A . . . . . 1.44 Megabyte (3.5
inch)

Graphics Mode . . . . . 03 (80-Column Text)

Primary Monitor attached to . . . . . Cirrus CL-GD5446
Graphics Controller . . . . . with Video Graphics

Color Monitor

Total Video Memory . . . . . 1024 Kbytes

Base Memory

```

```

System Total . . . . . 640 Kbytes
Amount Free . . . . . 597 Kbytes
(611888 Bytes)

Extended Memory
System Total . . . . . 392192 Kbytes

Expanded Memory
LIM Driver Support . . . . . LIM driver not
loaded

Operating System . . . . . MS-DOS version 7.00
(from diskette)

Status of file: C:\IO.SYS
Date and Time . . . . . 07/07/1997
17:16:26
Size . . . . . 0 Bytes

Status of file: C:\MSDOS.SYS
Date and Time . . . . . 07/07/1997
17:16:26
Size . . . . . 0 Bytes

-----
** Dump of C:\CONFIG.SYS
-----
** End of file

Status of file: C:\COMMAND.COM
*** File does not exist

-----
** Dump of C:\AUTOEXEC.BAT
-----
** End of file

Environment variables
PATH=
PROMPT=$P$G
COMSPEC=A:\COMMAND.COM
CMDLINE=inspect /u
End of environment

Revisions Table

Previous Revisions
I/O Board Revision . . . . . 02
Assembly Version . . . . . 1
Functional Revision Level . B

Current Revisions
I/O Board Revision . . . . . 02
Assembly Version . . . . . 1
Functional Revision Level . B

Memory Allocation (including INSPECT)
PSP SIZE NAME TRAPPED INTERRUPTS

```

```

-----
08D8 007200 COMMAND.COM F7h FDh 2Fh 2Eh 24h
23h 22h
0AA3 197856 INSPECT.EXE FBh F1h EFh EBh 3Fh
00h

System Configuration Memory
00 - 0F : 35 00 13 00 18 00 03 03 02 99 26
82 50 80 00 00
10 - 1F : 40 00 00 00 03 80 02 00 3C 00 00
00 00 00 00 02
20 - 2F : 00 00 00 00 7F 20 20 40 00 92 00
00 00 10 02 A4
30 - 3F : 00 3C 19 80 00 00 XX XX XX XX XX XX
XX XX XX XX XX

BIOS Data Area
40:0000 : F8 03 F8 02 00 00 00 00 BC 03 00
00 00 00 08 02
40:0010 : 27 44 00 80 02 00 00 00 00 00 1E
00 1E 00 00 00
40:0020 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:0030 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 01 01
40:0040 : 22 00 04 00 00 2E 01 0B 02 03 50
00 00 10 00 00
40:0050 : 00 14 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:0060 : 0E 0D 00 D4 03 29 30 C2 11 45 77
00 C2 37 12 00
40:0070 : 00 00 00 12 00 01 00 00 14 14 14
14 01 01 01 01
40:0080 : 1E 00 3E 00 18 10 00 60 F9 11 0B
01 00 00 00 05
40:0090 : 17 00 00 00 2E 00 10 00 00 00 00
00 00 00 00 00
40:00A0 : 00 00 00 00 00 00 00 00 42 54 00
C0 00 00 00 00
40:00B0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:00C0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:00D0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:00E0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:00F0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00

Interrupt Vector Table (including INSPECT)
00 - 03 : 0AB3:0555 0070:0465
080F:0016 0070:0465
04 - 07 : 0070:0465 F000:FF54
F000:93CC F000:9BD0
08 - 0B : 080F:001F 080F:0028
F000:9BD0 F000:9BD0
0C - 0F : F000:9BD0 F000:9BD0
080F:009A 0070:0465
10 - 13 : C000:3342 F000:F84D
F000:F841 0070:03EE

```

```

14 - 17 : F000:6E95 0249:0240
0070:042D F000:EPD2
18 - 1B : F000:BB31 08D0:002F
F000:FE6E 0070:045F
1C - 1F : F000:FF53 C000:1E2B
0000:0522 C000:6032
20 - 23 : 00C9:0FA8 00C9:0FB2
08D8:0314 08D8:016D
24 - 27 : 08D8:0178 00C9:0FBC
00C9:0FC6 00C9:0FD0
28 - 2B : 00C9:106F 0070:0466
00C9:106F 00C9:106F
2C - 2F : 00C9:106F 00C9:106F
08D8:0162 08D9:01CC
30 - 33 : C90F:E4EA F000:9B00
00C9:106F 00C9:106F
34 - 37 : 00C9:106F 00C9:106F
00C9:106F 00C9:106F
38 - 3B : 00C9:106F 00C9:106F
00C9:106F 00C9:106F
3C - 3F : 00C9:106F 00C9:106F
00C9:106F 1BBF:04F7
40 - 43 : F000:EC59 0000:0000
F000:F065 C000:5C32
44 - 47 : F000:9BD0 F000:9BD0
0000:0000 F000:9BD0
48 - 4B : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
4C - 4F : F000:9BD0 F000:9BD0
F000:9BD0 0070:04FC
50 - 53 : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
54 - 57 : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
58 - 5B : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
5C - 5F : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
60 - 63 : 0000:0000 0000:0000
0000:0000 0000:0000
64 - 67 : 0000:0000 0000:0000
0000:0000 0000:0000
68 - 6B : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
6C - 6F : F000:9BD0 C000:3342
F000:9BD0 F000:9BD0
70 - 73 : 080F:0035 F000:9C1F
080F:00B2 080F:00CA
74 - 77 : 080F:00E2 F000:9C28
080F:00FA F000:9BD0
78 - 7B : 0000:0000 0000:0000
0000:0000 0000:0000
7C - 7F : 0000:0000 0000:0000
0000:0000 0000:0000
80 - 83 : 0000:0000 0000:0000
0000:0000 0000:0000
84 - 87 : 0000:0000 0000:0000
0000:0000 0000:0000
88 - 8B : 0000:0000 0000:0000
0000:0000 0000:0000
8C - 8F : 0000:0000 0000:0000
0000:0000 0000:0000

```

```

90 - 93 : 0000:0000 0000:0000
0000:0000 0000:0000
94 - 97 : 0000:0000 0000:0000
0000:0000 0000:0000
98 - 9B : 0000:0000 0000:0000
0000:0000 0000:0000
9C - 9F : 0000:0000 0000:0000
0000:0000 0000:0000
A0 - A3 : 0000:0000 0000:0000
0000:0000 0000:0000
A4 - A7 : 0000:0000 0000:0000
0000:0000 0000:0000
A8 - AB : 0000:0000 0000:0000
0000:0000 0000:0000
AC - AF : 0000:0000 0000:0000
0000:0000 0000:0000
B0 - B3 : 0000:0000 0000:0000
0000:0000 0000:0000
B4 - B7 : 0000:0000 0000:0000
0000:0000 0000:0000
B8 - BB : 0000:0000 0000:0000
0000:0000 0000:0000
BC - BF : 0000:0000 0000:0000
0000:0000 0000:0000
C0 - C3 : 0000:0000 0000:0000
0000:0000 0000:0000
C4 - C7 : 0000:0000 0000:0000
0000:0000 0000:0000
C8 - CB : 0000:0000 0000:0000
0000:0000 0000:0000
CC - CF : 0000:0000 0000:0000
0000:0000 0000:0000
D0 - D3 : 0000:0000 0000:0000
0000:0000 0000:0000
D4 - D7 : 0000:0000 0000:0000
0000:0000 0000:0000
D8 - DB : 0000:0000 0000:0000
0000:0000 0000:0000
DC - DF : 0000:0000 0000:0000
0000:0000 0000:0000
E0 - E3 : 0000:0000 0000:0000
0000:0000 0000:0000
E4 - E7 : 0000:0000 0000:0000
0000:0000 0000:0000
E8 - EB : 0083:0000 0006:1FB9
0006:1FB7 0046:CE00
EC - EF : 0000:1DD3 0020:0203
0000:031F 0202:C000
F0 - F3 : C000:0008 0BE9:0E06
03CE:03DA 0008:5050
F4 - F7 : 00E8:655F 0203:0000
0000:03C4 0A2E:03C4
F8 - FB : 3394:0000 FF0E:000F
0000:0000 00F4:D3C7
FC - FF : 0246:0A00 0A00:0000
E0FB:0049 0003:0A1F

PCI Devices Information
Signature . . . . . PCI
Config Mechanism #1 . . . . Supported
Config Mechanism #2 . . . . Not Supported
Spec Cycle for Config #1 . . . Supported
Spec Cycle for Config #2 . . . Not Supported

```

```

BIOS Interface Version . . . . 2.10
Last PCI Bus Number . . . . . 1
Number of PCI Devices . . . . . 4

Bus Number . . . . . 0
Device Number . . . . . 11
Function Number . . . . . 00h
Slot Number . . . . . 0
Vendor ID . . . . . 1013h
Device ID . . . . . 00B8h
Revision ID . . . . . 45h
Device Type . . . . . VGA Compatible

Controller
Programming Interface . . . . 00h
Expansion ROM Base Address . . FFFF0000h
IRQ Line . . . . . 255
IRQ Pin . . . . . Not Used
Memory Address Base . . . . . C4000000h
Memory Address Length . . . . . 2000000h
Memory Address Base . . . . . C6EFF000h
Memory Address Length . . . . . 1000h

Bus Number . . . . . 1
Device Number . . . . . 7
Function Number . . . . . 00h
Slot Number . . . . . 0
Vendor ID . . . . . 0E11h
Device ID . . . . . AE43h
Revision ID . . . . . 10h
Device Type . . . . . Other Network

Controller
Programming Interface . . . . 00h
Expansion ROM Base Address . . FFFF0000h
IRQ Line . . . . . 9
IRQ Pin . . . . . INTA#
IO Address Base . . . . . 2800h
IO Address Length . . . . . 10h
Memory Address Base . . . . . C6FFDDF0h
Memory Address Length . . . . . 10h

Bus Number . . . . . 1
Device Number . . . . . 9
Function Number . . . . . 00h
Slot Number . . . . . 0
Vendor ID . . . . . 0E11h
Device ID . . . . . 000Fh
Revision ID . . . . . 14h
Device Type . . . . . SCSI Bus Controller
Programming Interface . . . . 00h
Expansion ROM Base Address . . 0h
IRQ Line . . . . . 10
IRQ Pin . . . . . INTA#
IO Address Base . . . . . 2000h
IO Address Length . . . . . 100h
Memory Address Base . . . . . C6FFDF00h
Memory Address Length . . . . . 100h
Memory Address Base . . . . . C6FFF000h
Memory Address Length . . . . . 1000h

Bus Number . . . . . 1
Device Number . . . . . 9
Function Number . . . . . 01h
Slot Number . . . . . 0

```

```

Vendor ID . . . . . 0E11h
Device ID . . . . . 000Fh
Revision ID . . . . . 14h
Device Type . . . . . SCSI Bus Controller
Programming Interface . . . . 00h
Expansion ROM Base Address . . 0h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTB#
IO Address Base . . . . . 2400h
IO Address Length . . . . . 100h
Memory Address Base . . . . . C6FFDE00h
Memory Address Length . . . . . 100h
Memory Address Base . . . . . C6FFE000h
Memory Address Length . . . . . 1000h

```

ProLiant 1600 is a trademark of Compaq Computer Corporation.

Server Registry

```

Key Name: SYSTEM\CurrentControlSet\Services
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Key Name: SYSTEM\CurrentControlSet\Services\Abiosdsk
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: Group
Type: REG_SZ
Data: Primary disk

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x3

```

```

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name: SYSTEM\CurrentControlSet\Services\Afd
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0

```

Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data:

Value 1
 Name: DisplayName
 Type: REG_SZ
 Data: AFD Networking Support Environment

Value 2
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 3
 Name: Group
 Type: REG_SZ
 Data: TDI

Value 4
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: \\SystemRoot\System32\drivers\afd.sys

Value 5
 Name: Start
 Type: REG_DWORD
 Data: 0x2

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Afd\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_AFD\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Afd\Linkage
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Afd\Linkage\Disable
 d
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Afd\Parameters
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Afd\Security
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: Security
 Type: REG_BINARY
 Data:
 00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
 00 18 00
 00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000050 00 00 00 05 00 00 1c 00 - fd 01 02 00 01
 02 00 00
 00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 12
 00 00 00#.....
 00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
 00 00 05
 00000080 20 00 00 00 20 02 00 00 - 12 00 00 00 00
 00 1c 00
 00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
 00 00 00
 000000a0 25 02 00 00 12 00 00 00 - 00 00 18 00 fd
 01 02 00 %.....
 000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
 02 00 00%...
 000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
 01 00 00
 000000d0 00 00 00 05 12 00 00 00 -

Key Name:
 SYSTEM\CurrentControlSet\Services\AFTRegistration
 Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: Compatible3
 Type: REG_SZ
 Data: Compaq,N100,qNFT,3601,32634

Key Name:
 SYSTEM\CurrentControlSet\Services\Ahal54x
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x6

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Ahal74x
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x8

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\aic78xx
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD

Data: 0x1
Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x1e

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Alerter
Class Name: <NO CLASS>
Last Write Time: 1/19/99 - 5:28 PM

Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data:

Value 1
Name: DependOnService
Type: REG_MULTI_SZ
Data: LanmanWorkstation

Value 2
Name: DisplayName
Type: REG_SZ
Data: Alerter

Value 3
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 4
Name: ImagePath
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\services.exe

Value 5
Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 6
Name: Start
Type: REG_DWORD
Data: 0x3

Value 7
Name: Type
Type: REG_DWORD
Data: 0x20

Key Name:
SYSTEM\CurrentControlSet\Services\Alerter\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_ALERTER\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Alerter\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Alerter\Linkage\Dis
abled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Alerter\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Value 0
Name: AlertNames
Type: REG_MULTI_SZ
Data:

Key Name:
SYSTEM\CurrentControlSet\Services\Alerter\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00

00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000050 00 00 00 00 00 00 1c 00 - fd 01 02 00 01
02 00 00
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 02
00 04 00#.....
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05
00000080 20 00 00 00 20 02 00 00 - 02 00 04 00 00
00 1c 00
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00
000000a0 25 02 00 00 02 00 04 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\Always
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x14

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ami0nt
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group

Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0xd

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\amsint
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x24

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Arrow
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x15

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\atapi
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: ImagePath
Type: REG_EXPAND_SZ
Data: System32\DRIVERS\atapi.sys

Value 3
Name: PlugPlayServiceType
Type: REG_DWORD
Data: 0x1

Value 4
Name: Start
Type: REG_DWORD
Data: 0

Value 5
Name: Tag
Type: REG_DWORD
Data: 0x19

Value 6
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\atapi\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ

Data: Root\LEGACY_ATAPI\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Atdisk
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:42 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 1
Name: Group
Type: REG_SZ
Data: Primary disk

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x1

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ati
Class Name: <NO CLASS>
Last Write Time: 1/6/99 - 1:40 PM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 1
Name: Group
Type: REG_SZ
Data: Video

Value 2
Name: ImagePath
Type: REG_EXPAND_SZ
Data: System32\DRIVERS\ati.sys

Value 3

```

Name: Start
Type: REG_DWORD
Data: 0x1

Value 4
Name: Tag
Type: REG_DWORD
Data: 0x1

Value 5
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ati\Device0
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:34 PM
Value 0
Name: AllowedPixelDepth
Type: REG_BINARY
Data: 00000000 00
.

Value 1
Name: AllowedRefreshRate
Type: REG_BINARY
Data: 00000000 00
.

Value 2
Name: AllowedResolution
Type: REG_BINARY
Data: 00000000 00
.

Value 3
Name: BlockWriteDisabled
Type: REG_DWORD
Data: 0

Value 4
Name: CustomCRTTimings
Type: REG_BINARY
Data: 00000000 00
.

Value 5
Name: DDC2Disabled
Type: REG_DWORD
Data: 0

Value 6
Name: DDC2Supported
Type: REG_BINARY
Data: 00000000 00 00 00 00
....

```

```

Value 7
Name: Device Description
Type: REG_SZ
Data: ATI Technologies Inc. 3D RAGE IIC

Value 8
Name: FastFillDisabled
Type: REG_DWORD
Data: 0

Value 9
Name: FileToRead
Type: REG_BINARY
Data: 00000000 5c 00 53 00 79 00 73 00 - 74 00 65 00 6d
00 52 00 \.S.y.s.t.e.m.R.
00000010 6f 00 6f 00 74 00 5c 00 - 53 00 79 00 73
00 74 00 o.o.t.\.S.y.s.t.
00000020 65 00 6d 00 33 00 32 00 - 5c 00 64 00 72
00 69 00 e.m.3.2.\.d.r.i.
00000030 76 00 65 00 72 00 73 00 - 5c 00 61 00 74
00 69 00 v.e.r.s.\.a.t.i.
00000040 76 00 64 00 69 00 66 00 - 2e 00 64 00 61
00 74 00 v.d.i.f...d.a.t.
00000050 00 00
..

Value 10
Name: HardwareInformation.AdapterString
Type: REG_BINARY
Data: 00000000 41 00 54 00 49 00 20 00 - 47 00 72 00 61
00 70 00 A.T.I. .G.r.a.p.
00000010 68 00 69 00 63 00 73 00 - 20 00 41 00 63
00 63 00 h.i.c.s. .A.c.c.
00000020 65 00 6c 00 65 00 72 00 - 61 00 74 00 6f
00 72 00 e.l.e.r.a.t.o.r.
00000030 00 00
..

Value 11
Name: HardwareInformation.BiosString
Type: REG_BINARY
Data: 00000000 53 00 47 00 52 00 41 00 - 4d 00 20 00 42
00 49 00 S.G.R.A.M. .B.I.
00000010 4f 00 53 00 20 00 33 00 - 2e 00 30 00 39
00 33 00 O.S. .3...0.9.3.
00000020 00 00
..

Value 12
Name: HardwareInformation.ChipType
Type: REG_BINARY
Data: 00000000 41 00 54 00 49 00 20 00 - 33 00 44 00 20
00 52 00 A.T.I. .3.D. .R.
00000010 41 00 47 00 45 00 20 00 - 49 00 49 00 43
00 20 00 A.G.E. .I.I.C. .
00000020 28 00 55 00 4d 00 43 00 - 20 00 47 00 54
00 2d 00 (.U.M.C. .G.T.-.

```

```

00000030 42 00 29 00 00 00
B.)...

Value 13
Name: HardwareInformation.Crc32
Type: REG_BINARY
Data: 00000000 02 d3 9f 7d
...}

Value 14
Name: HardwareInformation.DacType
Type: REG_BINARY
Data: 00000000 41 00 54 00 49 00 20 00 - 49 00 6e 00 74
00 65 00 A.T.I. .I.n.t.e.
00000010 72 00 6e 00 61 00 6c 00 - 20 00 44 00 41
00 43 00 r.n.a.l. .D.A.C.
00000020 00 00
..

Value 15
Name: HardwareInformation.MemorySize
Type: REG_BINARY
Data: 00000000 00 00 20 00
...

Value 16
Name: HDTVEnabled
Type: REG_DWORD
Data: 0

Value 17
Name: InstalledDisplayDrivers
Type: REG_MULTI_SZ
Data: ati

Value 18
Name: MonitorID
Type: REG_BINARY
Data: 00000000 00 00 00 00 00 00 47 00 - 65 00 6e 00 65
00 72 00 .....G.e.n.e.r.
00000010 69 00 63 00 00 00 00 00 - 00 00 00 00 00
00 00 00 i.c.....
00000020 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000030 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000040 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000050 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000060 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000070 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000080 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....

Value 19

```

Name: TimingSelection
 Type: REG_DWORD
 Data: 0x2

Value 20
 Name: TVOutAdjustments
 Type: REG_BINARY
 Data: 00000000 00

Value 21
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\ati\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM
 Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_ATI\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Beep
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: Base

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x2

Value 4
 Name: Type

Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Beep\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM
 Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_BEEP\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Browser
 Class Name: <NO CLASS>
 Last Write Time: 1/21/99 - 9:31 AM
 Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data:

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data: LanmanWorkstation
 LanmanServer
 LmHosts

Value 2
 Name: DisplayName
 Type: REG_SZ
 Data: Computer Browser

Value 3
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\services.exe

Value 5
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 6
 Name: Start

Type: REG_DWORD
 Data: 0x3

Value 7
 Name: Type
 Type: REG_DWORD
 Data: 0x20

Key Name:
 SYSTEM\CurrentControlSet\Services\Browser\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM
 Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_BROWSER\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Browser\Linkage
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Browser\Linkage\Dis
 abled
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Browser\Parameters
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: IsDomainMaster
 Type: REG_SZ
 Data: FALSE

Value 1
 Name: MaintainServerList
 Type: REG_SZ
 Data: Auto

Key Name:
 SYSTEM\CurrentControlSet\Services\Browser\Security
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: Security
 Type: REG_BINARY
 Data:


```

00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 01 08 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 02
00 20 00 .....#.....
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 02 00 20 00 00
00 1c 00 .....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 02 00 20 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\BusLogic
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

```

```

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x1b

```

```

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Busmouse
Class Name: <NO CLASS>

```

```
Last Write Time: 10/10/96 - 2:09 AM
```

```

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: Group
Type: REG_SZ
Data: Pointer Port

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x3

```

```

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Busmouse\Parameters
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
Name: MouseDataQueueSize
Type: REG_DWORD
Data: 0x64

```

```

Value 1
Name: NumberOfButtons
Type: REG_DWORD
Data: 0x2

```

```

Value 2
Name: PointerDeviceBaseName
Type: REG_SZ
Data: PointerPort

```

```

Value 3
Name: SampleRate
Type: REG_DWORD
Data: 0x32

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Cdaudio
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: Group

```

```

Type: REG_SZ
Data: Filter

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x1

```

```

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x6

```

```

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Cdaudio\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

```

```

Value 0
Name: Count
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: NextInstance
Type: REG_DWORD
Data: 0

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Cdfs
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data: SCSI CDROM Class

```

```

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

```

```

Value 2
Name: Group
Type: REG_SZ
Data: File system

```

```

Value 3
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 4
Name: Type
Type: REG_DWORD
Data: 0x2

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Cdfs\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_CDFS\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Cdrom
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: Autorun
Type: REG_DWORD
Data: 0x1

Value 1
Name: DependOnGroup
Type: REG_MULTI_SZ
Data: SCSI miniport

Value 2
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 3
Name: Group
Type: REG_SZ
Data: SCSI CDROM Class

Value 4
Name: Start
Type: REG_DWORD
Data: 0x1

Value 5
Name: Tag
Type: REG_DWORD
Data: 0x2

Value 6
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Cdrom\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_CDROM\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Changer
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 1
Name: Group
Type: REG_SZ
Data: Filter

Value 2
Name: Start
Type: REG_DWORD
Data: 0x1

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x5

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Changer\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: Count
Type: REG_DWORD
Data: 0

Value 1
Name: NextInstance
Type: REG_DWORD
Data: 0

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\cirrus
Class Name: <NO CLASS>
Last Write Time: 1/6/99 - 1:40 PM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: Video

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\cirrus\Device0
Class Name: <NO CLASS>
Last Write Time: 1/5/99 - 4:35 PM
Value 0
Name: HardwareInformation.AdapterString
Type: REG_BINARY
Data:
00000000 43 00 69 00 72 00 72 00 - 75 00 73 00 20
00 4c 00 C.i.r.r.u.s. .L.
00000010 6f 00 67 00 69 00 63 00 - 20 00 43 00 6f
00 6d 00 o.g.i.c. .C.o.m.
00000020 70 00 61 00 74 00 69 00 - 62 00 6c 00 65
00 00 00 p.a.t.i.b.l.e...

Value 1
Name: HardwareInformation.ChipType
Type: REG_BINARY
Data:
00000000 43 00 69 00 72 00 72 00 - 75 00 73 00 20
00 4c 00 C.i.r.r.u.s. .L.
00000010 6f 00 67 00 69 00 63 00 - 20 00 35 00 34
00 34 00 o.g.i.c. .5.4.4.
00000020 36 00 00 00
6...

Value 2
Name: HardwareInformation.Crc32
Type: REG_BINARY
Data:
00000000 57 d7 f0 ab
W...

Value 3
Name: HardwareInformation.DacType
Type: REG_BINARY

```

Data:
 00000000 49 00 6e 00 74 00 65 00 - 67 00 72 00 61
 00 74 00 I.n.t.e.g.r.a.t.
 00000010 65 00 64 00 20 00 52 00 - 41 00 4d 00 44
 00 41 00 e.d. .R.A.M.D.A.
 00000020 43 00 00 00
 C...

Value 4
 Name: HardwareInformation.MemorySize
 Type: REG_BINARY
 Data: 00 00 20 00
 ..

Value 5
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: vga
 cirrus
 vga256
 vga64K

Value 6
 Name: RequestedSystemResources
 Type: REG_RESOURCE_REQUIREMENTS_LIST
 Data:
 Interface Type: Internal
 Bus Number: 0
 Slot Number: 0
 List 0

Descriptor 0
 Resource: Memory
 Option: 0x00000001
 Disposition: Device

Exclusive

Type: Write Only
 Length: 0x2000000
 Alignment: 0x2000000
 Minimum Address: 0xf4000000
 Maximum Address: 0xf5ffffff

Descriptor 1
 Resource: Memory
 Option: 0x00000009
 Disposition: Device

Exclusive

Type: Write Only
 Length: 0x2000000
 Alignment: 0x2000000
 Minimum Address: 0xf4000000
 Maximum Address: 0xf5ffffff

Descriptor 2
 Resource: Memory
 Option: 0x00000008
 Disposition: Device

Exclusive

Type: Write Only
 Length: 0x2000000
 Alignment: 0x2000000
 Minimum Address: 0xf74a0000

Maximum Address: 0xf74a0fff

Descriptor 3
 Resource: Memory
 Option: 0x00000001
 Disposition: Device

Exclusive

Type: Read / Write
 Length: 0x1000
 Alignment: 0x1000
 Minimum Address: 0xf74a0000
 Maximum Address: 0xf74a0fff

Descriptor 4
 Resource: Memory
 Option: 0x00000008
 Disposition: Device

Exclusive

Type: Read / Write
 Length: 0x1000
 Alignment: 0x1000
 Minimum Address: 0xf74a0000
 Maximum Address: 0xf74a0fff

Value 7
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\cirrus\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_CIRRUS\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\ClipSrv
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 6:06 AM

Value 0
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data: NetDDE

Value 1

Name: DisplayName
 Type: REG_SZ
 Data: ClipBook Server

Value 2
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 3
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\system32\clipsrv.exe

Value 4
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 5
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x10

Key Name:
 SYSTEM\CurrentControlSet\Services\ClipSrv\Security
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: Security
 Type: REG_BINARY
 Data:
 00000000 01 00 14 80 9c 00 00 00 - a8 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000030 20 02 00 00 02 00 68 00 - 04 00 00 00 00
 03 18 00h.....
 00000040 8d 00 02 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
 02 00 00
 00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
 03 18 00
 00000070 8f 00 02 00 01 02 00 00 - 00 00 00 05 20
 00 00 00
 00000080 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
 01 00 00 #.....
 00000090 00 00 00 05 04 00 00 00 - 23 02 00 00 01
 01 00 00#.
 000000a0 00 00 00 05 12 00 00 00 - 01 01 00 00 00
 00 00 05
 000000b0 12 00 00 00

<p>Key Name: SYSTEM\CurrentControlSet\Services\Cpqarray Class Name: <NO CLASS> Last Write Time: 12/11/98 - 5:47 AM</p> <p>Value 0 Name: ErrorControl Type: REG_DWORD Data: 0x1</p> <p>Value 1 Name: Group Type: REG_SZ Data: SCSI miniport</p> <p>Value 2 Name: ImagePath Type: REG_EXPAND_SZ Data: System32\DRIVERS\cpqarray.sys</p> <p>Value 3 Name: PlugPlayServiceType Type: REG_DWORD Data: 0x1</p> <p>Value 4 Name: Start Type: REG_DWORD Data: 0</p> <p>Value 5 Name: Tag Type: REG_DWORD Data: 0x100</p> <p>Value 6 Name: Type Type: REG_DWORD Data: 0x1</p> <p>Key Name: SYSTEM\CurrentControlSet\Services\cpqfws2e Class Name: <NO CLASS> Last Write Time: 10/10/96 - 2:09 AM</p> <p>Value 0 Name: ErrorControl Type: REG_DWORD Data: 0x1</p> <p>Value 1 Name: Group Type: REG_SZ Data: SCSI miniport</p> <p>Value 2 Name: Start Type: REG_DWORD Data: 0x4</p> <p>Value 3 Name: Tag Type: REG_DWORD</p>	<p>Data: 0x26</p> <p>Value 4 Name: Type Type: REG_DWORD Data: 0x1</p> <p>Key Name: SYSTEM\CurrentControlSet\Services\CPQSET Class Name: GenericClass Last Write Time: 1/29/99 - 6:48 PM</p> <p>Key Name: SYSTEM\CurrentControlSet\Services\CPQSET\Adapters Class Name: GenericClass Last Write Time: 2/2/99 - 11:54 AM</p> <p>Value 0 Name: CP100 Type: REG_SZ Data: Compaq</p> <p>Key Name: SYSTEM\CurrentControlSet\Services\CPQSET\CP100 Class Name: GenericClass Last Write Time: 2/2/99 - 11:54 AM</p> <p>Value 0 Name: AdapterDescription Type: REG_SZ Data: DIAG100_GetAdapterDescription</p> <p>Value 1 Name: Configure Type: REG_SZ Data: DIAG100_Configure</p> <p>Value 2 Name: Detect Type: REG_SZ Data: DIAG100_Detect</p> <p>Value 3 Name: DeviceExist Type: REG_SZ Data: DIAG100_DeviceExist</p> <p>Value 4 Name: Diagnose Type: REG_SZ Data: DIAG100_Diagnose</p> <p>Value 5 Name: DLL Type: REG_SZ Data: CP100.DLL</p> <p>Value 6 Name: GetExtendedFeatures Type: REG_SZ Data: DIAG100_GetExtendedFeatures</p> <p>Value 7</p>	<p>Name: Help Type: REG_SZ Data: N100SET.HLP</p> <p>Value 8 Name: InstallAnyway Type: REG_SZ Data: DIAG100_InstallAnyway</p> <p>Value 9 Name: RegistryKey Type: REG_SZ Data: DIAG100_GetRegistryKey</p> <p>Value 10 Name: Summary Type: REG_SZ Data: DIAG100_Resource_Summary</p> <p>Key Name: SYSTEM\CurrentControlSet\Services\CPQSET\CP100\Parameters Class Name: GenericClass Last Write Time: 2/2/99 - 11:54 AM</p> <p>Value 0 Name: Adaptive_IFS Type: REG_SZ Data: 1,7,Adaptive Inter-Frame Spacing,0,2,1,0,255,1</p> <p>Value 1 Name: BusNumber Type: REG_SZ Data: 0,7,BusNumber,0,2,0,0,16,1</p> <p>Value 2 Name: BusType Type: REG_SZ Data: 0,7,BusType,0,2,5,2,5,1</p> <p>Value 3 Name: BusTypeLocal Type: REG_SZ Data: 0,7,BusTypeLocal,0,2,5,2,5,1</p> <p>Value 4 Name: Eid Type: REG_SZ Data: 0,7,Eid,0,2,0,0,4294967295,1</p> <p>Value 5 Name: Fifo Type: REG_SZ Data: 0,3,Fifo Depth,0,2,12,0,15,1</p> <p>Value 6 Name: ForceDpx Type: REG_SZ Data: 1,4,Duplex Mode,0,1,Auto,Auto,Half,Full</p> <p>Value 7</p>
---	---	--

Name: MapRegisters
 Type: REG_SZ
 Data:

Value 8
 Name: MediaType
 Type: REG_SZ
 Data: 0,7,MediaType,0,2,1,1,1,1

Value 9
 Name: MsPciScan
 Type: REG_SZ
 Data: 0,4,MsPciScan,0,2,1,0,1,1

Value 10
 Name: NetworkAddress
 Type: REG_SZ
 Data: 1,7,Locally Administered
 Address,0,5,0,0,1,1

Value 11
 Name: NumCoalesce
 Type: REG_SZ
 Data: 1,7,Coalesce Buffers,0,2,8,1,32,1

Value 12
 Name: NumRfd
 Type: REG_SZ
 Data: 1,7,Receive
 Buffers,0,2,32,1,1024,1

Value 13
 Name: NumTbd
 Type: REG_SZ
 Data: 0,3,Transmit Buffer
 Descriptors,0,2,64,1,65535,1

Value 14
 Name: NumTbdPerTcb
 Type: REG_SZ
 Data: 0,4,Transmit Buffers per
 Frame,0,2,12,1,16,1

Value 15
 Name: NumTcb
 Type: REG_SZ
 Data: 1,7,Transmit Control
 Blocks,0,2,16,1,80,1

Value 16
 Name: Off
 Type: REG_SZ
 Data: 1,3,Off Timer,0,2,2,1,65535,1

Value 17
 Name: On
 Type: REG_SZ
 Data: 1,3,On Timer,0,2,32768,1,65535,1

Value 18
 Name: PerfOptims
 Type: REG_SZ
 Data: 0,4,PerfOptims,0,2,0,0,65535,1

Value 19
 Name: RxDmaCount
 Type: REG_SZ
 Data: 0,4,RxDmaCount,0,2,0,0,63,1

Value 20
 Name: RxFifo
 Type: REG_SZ
 Data: 0,4,Receive Fifo
 Depth,0,2,8,0,15,1

Value 21
 Name: Slot
 Type: REG_SZ
 Data:

Value 22
 Name: Speed
 Type: REG_SZ
 Data: 1,7,Network
 Speed,0,4,Auto,Auto,0,10Mbps,10,100Mbps,100

Value 23
 Name: Threshold
 Type: REG_SZ
 Data: 0,7,Transmit
 Threshold,0,2,16,0,200,1

Value 24
 Name: TxDmaCount
 Type: REG_SZ
 Data: 0,4,TxDmaCount,0,2,0,0,63,1

Value 25
 Name: TxFifo
 Type: REG_SZ
 Data: 0,4,Transmit Fifo
 Depth,0,2,8,0,15,1

Value 26
 Name: Txmitwait
 Type: REG_SZ
 Data: 0,7,Txmitwait,0,2,1,0,255,1

Value 27
 Name: UcodeSW
 Type: REG_SZ
 Data: 0,7,UcodeSW,0,2,1,0,1,1

Value 28
 Name: UnderrunRetry
 Type: REG_SZ
 Data: 0,4,UnderrunRetry,0,2,1,0,3,1

Key Name:
 SYSTEM\CurrentControlSet\Services\CpqSMRT2
 Class Name: <NO CLASS>
 Last Write Time: 1/5/99 - 2:16 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD

Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: Primary disk

Value 2
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: System32\drivers\CPQSMRT2.SYS

Value 3
 Name: PlugPlayServiceType
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Start
 Type: REG_DWORD
 Data: 0

Value 5
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\CpqSMRT2\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\SCSIADAPTER\OEM0.INF&CPQSMRT2

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\CpqSMRT2\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 1/5/99 - 2:20 PM

Value 0
 Name: InterruptMask
 Type: REG_DWORD
 Data: 0xffffffff

Key Name:
 SYSTEM\CurrentControlSet\Services\CpqSMRT2\Security
 Class Name: <NO CLASS>
 Last Write Time: 1/5/99 - 2:16 PM
 Value 0
 Name: Security
 Type: REG_BINARY
 Data:
 00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
 00 18 00
 00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000050 b0 ca 00 08 00 00 1c 00 - fd 01 02 00 01
 02 00 00
 00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 c7
 a4 61 73#.....as
 00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
 00 00 05
 00000080 20 00 00 00 20 02 00 00 - c7 a4 61 73 00
 00 1c 00as.....
 00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
 00 00 00
 000000a0 25 02 00 00 c7 a4 61 73 - 00 00 18 00 fd
 01 02 00 %.....as.....
 000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
 02 00 00%.....
 000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
 01 00 00
 000000d0 00 00 00 05 12 00 00 00 -

Key Name:
 SYSTEM\CurrentControlSet\Services\dac960nt
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1
 Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport
 Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4
 Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x20
 Value 4

Name: Type
 Type: REG_DWORD
 Data: 0x1
 Key Name:
 SYSTEM\CurrentControlSet\Services\dce376nt
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1
 Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport
 Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4
 Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x16
 Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Dell_DGX
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM
 Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0
 Value 1
 Name: Group
 Type: REG_SZ
 Data: Video
 Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4
 Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1
 Key Name:
 SYSTEM\CurrentControlSet\Services\Dell_DGX\Device0
 Class Name: <NO CLASS>

Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: framebuffer

Value 1
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\Delldsa
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x101

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP
 Class Name: <NO CLASS>
 Last Write Time: 2/8/99 - 10:11 AM
 Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data:

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data: Tcpip
 Afd
 NetBT

Value 2
 Name: DisplayName
 Type: REG_SZ

Data: DHCP Client

Value 3
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Group
 Type: REG_SZ
 Data: TDI

Value 5
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\services.exe

Value 6
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 7
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 8
 Name: Type
 Type: REG_DWORD
 Data: 0x20

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Linkage
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Linkage\Disabled
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options\1
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: KeyType
 Type: REG_DWORD
 Data: 0x1

Data: 0x7

Value 1
 Name: RegLocation
 Type: REG_SZ
 Data: System\CurrentControlSet\Services\?\Parameters\Tcpip\DhcpSubnetMaskOpt

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options\15
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: KeyType
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: RegLocation
 Type: REG_SZ
 Data: System\CurrentControlSet\Services\Tcpip\Parameters\DhcpDomain

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options\3
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: KeyType
 Type: REG_DWORD
 Data: 0x7

Value 1
 Name: RegLocation
 Type: REG_SZ
 Data: System\CurrentControlSet\Services\?\Parameters\Tcpip\DhcpDefaultGateway

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options\44
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: KeyType
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: RegLocation
 Type: REG_SZ
 Data: System\CurrentControlSet\Services\NetBT\Adapters\?\DhcpNameServer

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options\46
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: KeyType
 Type: REG_DWORD
 Data: 0x4

Value 1
 Name: RegLocation
 Type: REG_SZ
 Data: System\CurrentControlSet\Services\NetBT\Parameters\DhcpNodeType

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options\47
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: KeyType
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: RegLocation
 Type: REG_SZ
 Data: System\CurrentControlSet\Services\NetBT\Parameters\DhcpScopeID

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Parameters\Options\6
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: KeyType
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: RegLocation
 Type: REG_SZ
 Data: System\CurrentControlSet\Services\Tcpip\Parameters\DhcpNameServer

Key Name:
 SYSTEM\CurrentControlSet\Services\DHCP\Security
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: Security
 Type: REG_BINARY
 Data:

```

00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 78 96 14 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 x.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 8d
00 02 00 .....#.....
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 8d 00 02 00 00
00 1c 00 .....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 8d 00 02 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Disk
Class Name: <NO CLASS>
Last Write Time: 1/5/99 - 2:16 PM
Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data: SCSI miniport

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 2
Name: Group
Type: REG_SZ
Data: SCSI Class

Value 3
Name: Start
Type: REG_DWORD
Data: 0

Value 4
Name: Tag
Type: REG_DWORD
Data: 0x2

Value 5
Name: Type
Type: REG_DWORD

```

```

Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Disk\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_DISK\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Diskperf
Class Name: <NO CLASS>
Last Write Time: 1/22/99 - 7:44 PM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: Filter

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x4

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Diskperf\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_DISKPERF\0000

Value 1

```

```

Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\DptScsi
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x9

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\dtc329x
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x17

```


Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\et4000
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\et4000\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: vga
 w32
 vga256
 vga64K

Value 1
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 6:06 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ

Data: Event log

Value 2
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\system32\services.exe

Value 3
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 4
 Name: PlugPlayServiceType
 Type: REG_DWORD
 Data: 0x3

Value 5
 Name: Start
 Type: REG_DWORD
 Data: 0x2

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x20

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Application
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:34 PM

Value 0
 Name: File
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\system32\config\AppEvent.Evt

Value 1
 Name: MaxSize
 Type: REG_DWORD
 Data: 0x80000

Value 2
 Name: Retention
 Type: REG_DWORD
 Data: 0

Value 3
 Name: Sources
 Type: REG_MULTI_SZ
 Data: Winlogon
 Windows 3.1 Migration
 VBRuntime
 Userenv
 Type 1 Installer
 SQLServerProfiler
 SQLServerAgent
 SQLCTR70
 Replicator
 Perfmon
 Perflib

Perfctr
 ntbackup
 MSSQLServer
 MssCi
 MSDTC
 Microsoft Search
 LicenseService
 Java VM
 hpmon
 DrWatson
 DataTransformationServices
 Autochk
 Application

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Application\Autochk
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\winlogon.exe

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Application\DataTransformationServices
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:34 PM

Value 0
 Name: ErrorMessageFile
 Type: REG_SZ
 Data: C:\MSSQL7\Binn\dtspkg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Application\DrWatson
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\drwtsn32.exe

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\hpmo
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\hpmo.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\Java VM
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:16 PM
Value 0
Name: EventMessageFile
Type: REG_SZ
Data: C:\WINNT40\System32\vmhelper.dll

Value 1
Name: TypesSupported
Type: REG_BINARY
Data: 00000000 07 00 00 00
....

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\LicenseService
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: EventMessageFile
Type: REG_SZ
Data: %SystemRoot%\System32\llsrpc.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\Microsoft Search
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM
Value 0
Name: CategoryCount
Type: REG_DWORD
Data: 0x4

Value 1
Name: CategoryMessageFile

Type: REG_SZ
Data: C:\Program Files\Common
Files\System\MSSearch\bin\mssmsg.dll

Value 2
Name: EventMessageFile
Type: REG_SZ
Data: C:\Program Files\Common
Files\System\MSSearch\bin\mssmsg.dll

Value 3
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\MSDTC
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM
Value 0
Name: CategoryCount
Type: REG_DWORD
Data: 0x6

Value 1
Name: CategoryMessageFile
Type: REG_EXPAND_SZ
Data: C:\WINNT40\System32\ENUdte.dll

Value 2
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: C:\WINNT40\System32\ENUdte.dll

Value 3
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\MssCi
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM
Value 0
Name: CategoryCount
Type: REG_DWORD
Data: 0x4

Value 1
Name: CategoryMessageFile
Type: REG_SZ
Data: C:\Program Files\Common
Files\system\ContentIndex\oquery.dll

Value 2
Name: EventMessageFile
Type: REG_SZ
Data: C:\Program Files\Common
Files\system\ContentIndex\oquery.dll

Value 3
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\MSSQLServer
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:34 PM
Value 0
Name: CategoryCount
Type: REG_DWORD
Data: 0x8

Value 1
Name: CategoryMessageFile
Type: REG_SZ
Data: C:\MSSQL7\BINN\SQLLEVN70.DLL

Value 2
Name: EventMessageFile
Type: REG_SZ
Data: C:\MSSQL7\BINN\SQLLEVN70.DLL

Value 3
Name: TypesSupported
Type: REG_DWORD
Data: 0xff

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\ntbackup
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\ntbackup.exe

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Appl
n\Perfctrs
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\perfctrs.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\Perflib
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\prflbmsg.dll
 Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\Perfmon
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\perfmon.exe
 Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\Replicator
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\NetMessage.dll
 Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\SQLCTR70
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:34 PM
 Value 0
 Name: EventMessageFile
 Type: REG_SZ
 Data: C:\MSSQL7\BINN\SQLLEVN70.DLL
 Value 1
 Name: TypesSupported
 Type: REG_DWORD

Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\SQLServerAgent
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:34 PM
 Value 0
 Name: CategoryCount
 Type: REG_DWORD
 Data: 0x4
 Value 1
 Name: CategoryMessageFile
 Type: REG_SZ
 Data: C:\MSSQL7\BINN\SQLAGENT.DLL
 Value 2
 Name: EventMessageFile
 Type: REG_SZ
 Data: C:\MSSQL7\BINN\SQLAGENT.DLL
 Value 3
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\SQLServerProfiler
 Class Name: <NO CLASS>
 Last Write Time: 1/5/99 - 5:55 PM
 Value 0
 Name: CategoryCount
 Type: REG_DWORD
 Data: 0x8
 Value 1
 Name: CategoryMessageFile
 Type: REG_EXPAND_SZ
 Data: C:\MSSQL7\BINN\COMNEVNT.DLL
 Value 2
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: C:\MSSQL7\BINN\COMNEVNT.DLL
 Value 3
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\Type 1 Installer
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ

Data: %SystemRoot%\System32\tlinstal.dll
 Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\Userenv
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\userenv.dll
 Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\VBRunTime
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:34 PM
 Value 0
 Name: EventMessageFile
 Type: REG_SZ
 Data: C:\WINNT40\System32\MSVBVM60.DLL
 Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x4
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\Windows 3.1 Migration
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\advapi32.dll
 Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7
 Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Applicatio
 n\Winlogon
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile

Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\winlogon.exe

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:34 PM

Value 0
Name: File
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\config\SecEvent.Evt

Value 1
Name: MaxSize
Type: REG_DWORD
Data: 0x80000

Value 2
Name: PrimaryModule
Type: REG_SZ
Data: Security

Value 3
Name: Retention
Type: REG_DWORD
Data: 0

Value 4
Name: Sources
Type: REG_MULTI_SZ
Data: Spooler
Security Account Manager
SC Manager
NetDDE Object
LSA
Security

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\LSA
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ParameterMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\MsObjs.dll

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\LSA\ObjectNames
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0

Name: PolicyObject
Type: REG_DWORD
Data: 0x1600

Value 1
Name: SecretObject
Type: REG_DWORD
Data: 0x1610

Value 2
Name: TrustedDomainObject
Type: REG_DWORD
Data: 0x1620

Value 3
Name: UserAccountObject
Type: REG_DWORD
Data: 0x1630

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\NetDDE Object
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ParameterMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\MsObjs.dll

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\NetDDE Object\ObjectNames
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: DDE Share
Type: REG_DWORD
Data: 0x1d00

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\SC Manager
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ParameterMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\MsObjs.dll

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\SC Manager\ObjectNames
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: SC_MANAGER Object
Type: REG_DWORD
Data: 0x1c00

Value 1
Name: SERVICE Object
Type: REG_DWORD
Data: 0x1c10

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\Secur
ity
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: CategoryCount
Type: REG_DWORD
Data: 0x7

Value 1
Name: CategoryMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\MsAuditE.dll

Value 2
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\MsAuditE.dll

Value 3
Name: ParameterMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\MsObjs.dll

Value 4
Name: TypesSupported
Type: REG_DWORD
Data: 0x1c

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\Security\Secur
ity\ObjectNames
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Channel
Type: REG_DWORD
Data: 0x1400

Value 1
Name: Desktop
Type: REG_DWORD
Data: 0x1a10

Value 2
Name: Device
Type: REG_DWORD
Data: 0x1100

Value 3
Name: Directory
Type: REG_DWORD
Data: 0x1110

Value 4

Name: Event
 Type: REG_DWORD
 Data: 0x1120

Value 5
 Name: EventPair
 Type: REG_DWORD
 Data: 0x1130

Value 6
 Name: File
 Type: REG_DWORD
 Data: 0x1140

Value 7
 Name: IoCompletion
 Type: REG_DWORD
 Data: 0x1300

Value 8
 Name: Key
 Type: REG_DWORD
 Data: 0x1150

Value 9
 Name: MailSlot
 Type: REG_DWORD
 Data: 0x1140

Value 10
 Name: Mutant
 Type: REG_DWORD
 Data: 0x1160

Value 11
 Name: NamedPipe
 Type: REG_DWORD
 Data: 0x1140

Value 12
 Name: Port
 Type: REG_DWORD
 Data: 0x1170

Value 13
 Name: Process
 Type: REG_DWORD
 Data: 0x1180

Value 14
 Name: Profile
 Type: REG_DWORD
 Data: 0x1190

Value 15
 Name: Section
 Type: REG_DWORD
 Data: 0x11a0

Value 16
 Name: Semaphore
 Type: REG_DWORD
 Data: 0x11b0

Value 17
 Name: SymbolicLink
 Type: REG_DWORD
 Data: 0x11c0

Value 18
 Name: Thread
 Type: REG_DWORD
 Data: 0x11d0

Value 19
 Name: Timer
 Type: REG_DWORD
 Data: 0x11e0

Value 20
 Name: Token
 Type: REG_DWORD
 Data: 0x11f0

Value 21
 Name: Type
 Type: REG_DWORD
 Data: 0x1200

Value 22
 Name: WindowStation
 Type: REG_DWORD
 Data: 0x1a00

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Security\Secur
 ity Account Manager
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ParameterMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\MsObjs.dll

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Security\Secur
 ity Account Manager\ObjectNames
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: SAM_ALIAS
 Type: REG_DWORD
 Data: 0x1530

Value 1
 Name: SAM_DOMAIN
 Type: REG_DWORD
 Data: 0x1510

Value 2
 Name: SAM_GROUP
 Type: REG_DWORD
 Data: 0x1520

Value 3
 Name: SAM_SERVER
 Type: REG_DWORD
 Data: 0x1500

Value 4
 Name: SAM_USER
 Type: REG_DWORD
 Data: 0x1540

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Security\Secur
 ity Account Manager
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ParameterMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\MsObjs.dll

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\Security\Secur
 ity Account Manager\ObjectNames
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: Document
 Type: REG_DWORD
 Data: 0x1520

Value 1
 Name: Printer
 Type: REG_DWORD
 Data: 0x1510

Value 2
 Name: Server
 Type: REG_DWORD
 Data: 0x1500

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:34 PM

Value 0
 Name: File
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\system32\config\SysEvent.Evt

Value 1
 Name: MaxSize
 Type: REG_DWORD
 Data: 0x80000

Value 2
 Name: Retention
 Type: REG_DWORD
 Data: 0

Value 3
 Name: Sources
 Type: REG_MULTI_SZ
 Data: xga
 workstation
 weitekp9
 wdvga
 wd90c24a
 wd33c93
 VgaStart
 VgaSave
 v7vram
 UPS
 ultra24f
 ultra14f
 ultra124
 tmv1
 tdi
 Tcpip
 tl3b
 tl28
 SysMgmt
 symc810
 Srv
 spock
 sparrow
 sndblst
 slcd32
 Simbad
 sfloppy
 Service Control Manager
 server
 sermouse
 serial
 scsiscan
 scsiprnt
 scsiport
 Schedule
 SCache
 Save Dump
 SAM
 s3
 Rdr
 qv
 ql10wnt
 psidisp
 Print
 pnpisa
 pcmcia
 parvdm
 parport
 parallel
 oliscsi
 null
 NtServicePack
 ntfs
 npfs
 NetLogon
 NetDDE
 NetBT
 NetBIOS
 ndis
 nrcr710

nrcr700
 ncr77c22
 ncr53c9x
 Nbf
 N100
 Mup
 msfs
 msadlib
 mouclass
 Modem
 mkecr5xx
 mitsumi
 mga_mil
 mga
 LmHosts
 kbdclass
 Jzvx1484
 JazzSnd
 JazzG364
 JazzG300
 inport
 i8042prt
 ftdisk
 fs_rec
 floppy
 flashpnt
 fd8XX
 fd7000ex
 fd16_700
 fastfat
 eventlog
 et4000
 dtc329x
 dptscsi
 diskperf
 disk
 DHCP
 dell_dgx
 delldsa
 DCOM
 dce376nt
 dac960nt
 CpqSMRT2
 CPQSETUP
 CpqNF3
 cpqfws2e
 cpqarray
 cirrus
 changer
 cdrom
 cdfs
 cdaudio
 busmouse
 buslogic
 Browser
 beep
 ati
 atdisk
 atapi
 arrow
 Application Popup
 amsint
 ami0nt

always
 Alerter
 aic78xx
 ahal74x
 ahal54x
 Afd
 abiosdsk
 System

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\abi
 osdsk
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Afd
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\aha
 154x
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\aha
 174x
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\aic78xx
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\Alert
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\NetMessage.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\always
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\amiOnt
Class Name: <NO CLASS>

Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\amsint
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\Application Popup
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\ntdll.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\arrow
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\atapi

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\atdisk
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\atisk
Class Name: <NO CLASS>
Last Write Time: 1/6/99 - 1:40 PM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\drivers\ati.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\atp
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\Browser

Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\netevent.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\buslogic

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\busmouse

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\BusMouse.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\cdaudio

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD

Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\cds

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\cdrom

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\changer

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\cdrom

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported

Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\cpqarray

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\cpqfws2e

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\CpqNF3

Class Name: GenericClass
Last Write Time: 2/2/99 - 11:04 AM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\Netevent.dll;%SystemRoot%\System32\Drivers\NetFlx3.Sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\CPQSETUP

Class Name: <NO CLASS>
Last Write Time: 1/6/99 - 1:39 PM

Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: C:\WINNT40\SYSTEM32\IOLOGMSG.DLL

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\CpqSMRT2
 Class Name: 33554432
 Last Write Time: 1/5/99 - 2:16 PM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\Drivers\CQPSMRT2.SYS;%SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\dac960nt
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\dce376nt
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\DCOM
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0

Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: ParameterMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\kernel32.dll

Value 2
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\del1_dgx
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\del1dsa
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\DHP
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\dhcpcsvc.dll

Value 1
 Name: ParameterMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\kernel32.dll

Value 2

Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\disk
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\diskperf
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\dptscsi
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\dtc329x
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\et4000
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\eventlog
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\fastfat
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\fd16_700
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\fd700ex
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\fd8XX
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\fla shpnt
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\floppy
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ

Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\fs_rec
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\ftdisk
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\FtDisk.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\i8042prt
 Class Name: <NO CLASS>
 Last Write Time: 1/8/99 - 11:42 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\drivers\i8042prt.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\input

Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\InPort.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Jaz
 zG300
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Jaz
 zG364
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Jaz
 zSnd
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Jzv
 x1484
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\kbd
 class
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 2:56 PM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\drivers\kbdclass.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\LmH
 osts
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\mga
 mil
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\mga.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\mga
 mil
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\mga_mil.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\mit
 sumi
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\mke
 cr5xx
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Mod
 em
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ

Data:
 %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\Modem.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\mouclass
 Class Name: <NO CLASS>
 Last Write Time: 1/8/99 - 11:42 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\drivers\mouclass.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\msadlib
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\mfs
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Mup
 Class Name: <NO CLASS>

Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\N10
 Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:54 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Nbf
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 3:17 PM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\ncr
 53c9x
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\ncr
 77c22
 Class Name: <NO CLASS>

Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\ncr
 c700
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\ncr
 c710
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\ncr
 s
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Net
 BIOS

Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\NETBT
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\NETDDE
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netdde.exe

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x1f

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\NetLogon
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\NetMessage.dll

Value 1
 Name: ParameterMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\Kernel32.dll

Value 2
 Name: TypesSupported
 Type: REG_DWORD

Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\npfs
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\ntfs
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\NtServicePack
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 3:40 PM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\spmsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\null
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported

Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\oliscsi
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\parallel
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\Parallel.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\parport
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\ParPort.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\parvdm
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: EventMessageFile

Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\ParVdm.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\pcmcia
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\Pcmcia.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\pnppisa
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\pnppisa.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\Print
 Class Name: <NO CLASS>
 Last Write Time: 1/19/99 - 4:55 PM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\LocalSpl.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\psidisp
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\ql10wnt
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\qv
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\qv.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\Rdr
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\s3
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\s3.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\SAM
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\samsrv.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\SaveDump
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\SaveDump.exe

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name: SYSTEM\CurrentControlSet\Services\EventLog\System\SCache
 Class Name: REG_EXPAND_SZ
 Last Write Time: 1/21/99 - 3:46 PM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\Drivers\SCache.sys

Value 1
 Name: TypesSupported
 Type: REG_DWORD

Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\Schedule
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\netevent.dll

Value 1
Name: ParameterMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\kernel32.dll

Value 2
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\scsiport
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\scsiport
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\scsiscan
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile

Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\serial
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\Serial.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\sermouse
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\SerMouse.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\server
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\netevent.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\Service Control Manager
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\netevent.dll

Value 1
Name: ParameterMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\kernel32.dll

Value 2
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\sflppy
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\Simbad
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\slcd32
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\snd
 blst
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\spa
 rrow
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\spo
 ck
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Srv
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\sym
 c810
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Sys
 Mgmt
 Class Name: <NO CLASS>
 Last Write Time: 1/6/99 - 1:39 PM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: C:\WINNT40\SYSTEM32\DRIVERS\SYSMGMT.SYS;C:\WINNT40\SY
 STEM32\IOLOGMSG.DLL

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\t12
 8
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\t13
 b
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile

Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\Tcp
 ip
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\netevent.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\tdi
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\tmv
 1
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
 Name: TypesSupported
 Type: REG_DWORD
 Data: 0x7

Key Name:
 SYSTEM\CurrentControlSet\Services\EventLog\System\ult
 ral24
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: EventMessageFile

Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\ultra14f
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\ultra24f
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\UPS
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\netmsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\v7vram
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile

Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\VgaSave
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\vga.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\VgaStart
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\vga.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\wd3c93
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\wd90c24a
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\wdvga
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\weitk9
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\weitk9.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\workstation
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: EventMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\NetMsg.dll

Value 1
Name: TypesSupported
Type: REG_DWORD

Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\EventLog\System\xga
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorMessageFile
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\IoLogMsg.dll;%SystemRoot%\System32\Drivers\xga.sys

Value 1
Name: TypesSupported
Type: REG_DWORD
Data: 0x7

Key Name:
SYSTEM\CurrentControlSet\Services\Fastfat
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: Boot file system

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Type
Type: REG_DWORD
Data: 0x2

Key Name:
SYSTEM\CurrentControlSet\Services\Fastfat\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_FASTFAT\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Fd16_700
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x10

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Fd7000ex
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0xe

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Fd8xx
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0xf

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Fd8xx\Parameters
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Fd8xx\Parameters\Device0
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: DriverParameter
Type: REG_SZ
Data: IRQ=5

Key Name:
SYSTEM\CurrentControlSet\Services\flashpnt
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD

Data: 0x4
Value 3
Name: Tag
Type: REG_DWORD
Data: 0x25

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Floppy
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 1
Name: Group
Type: REG_SZ
Data: Primary disk

Value 2
Name: Start
Type: REG_DWORD
Data: 0x1

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x2

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Floppy\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_FLOPPY\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Fs_Rec
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 1
Name: Group
Type: REG_SZ
Data: Boot file system

Value 2
Name: Start
Type: REG_DWORD
Data: 0x1

Value 3
Name: Type
Type: REG_DWORD
Data: 0x8

Key Name:
SYSTEM\CurrentControlSet\Services\Fs_Rec\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_FS_REC\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Ftdisk
Class Name: <NO CLASS>
Last Write Time: 1/27/99 - 4:20 PM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 1
Name: Group
Type: REG_SZ
Data: Filter

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x3

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\i8042prt
Class Name: <NO CLASS>
Last Write Time: 1/8/99 - 11:42 AM
Value 0
Name: DisplayName
Type: REG_SZ
Data: i8042 Keyboard and PS/2 Mouse Port Driver

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 2
Name: Group
Type: REG_SZ
Data: Keyboard Port

Value 3
Name: ImagePath
Type: REG_EXPAND_SZ
Data: System32\DRIVERS\i8042prt.sys

Value 4
Name: Start
Type: REG_DWORD
Data: 0x1

Value 5
Name: Tag
Type: REG_DWORD
Data: 0x2

Value 6
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\i8042prt\Description
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:33 PM
Value 0
Name: \Device\KeyboardPort0
Type: REG_BINARY
Data:

```

00000000 01 00 00 00 00 00 00 00 - 16 00 00 00 00
00 00 00 .....
00000010 20 00 00 00 00 00 00 00 -
.....

```

```

Value 1
Name:          \Device\PointerPort0
Type:          REG_BINARY
Data:
00000000 01 00 00 00 00 00 00 00 - 15 00 00 00 00
00 00 00 .....
00000010 1f 00 00 00 00 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\i8042prt\Enum
Class Name:    <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

```

```

Value 0
Name:          0
Type:          REG_SZ
Data:          Root\*\PNP030b\1_0_22_0_32_0

```

```

Value 1
Name:          1
Type:          REG_SZ
Data:          Root\*\PNP030b\1_15_22_0_32_0

```

```

Value 2
Name:          2
Type:          REG_SZ
Data:          Root\*\PNP0F03\1_0_21_0_31_0

```

```

Value 3
Name:          3
Type:          REG_SZ
Data:          Root\*\PNP0F12\1_0_21_0_31_0

```

```

Value 4
Name:          4
Type:          REG_SZ
Data:          Root\*\PNP0F12\1_15_21_0_31_0

```

```

Value 5
Name:          Count
Type:          REG_DWORD
Data:          0x5

```

```

Value 6
Name:          NextInstance
Type:          REG_DWORD
Data:          0x5

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\i8042prt\Parameters
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
Name:          EnableWheelDetection
Type:          REG_DWORD
Data:          0x1

```

```

Value 1
Name:          KeyboardDataQueueSize
Type:          REG_DWORD
Data:          0x64

```

```

Value 2
Name:          KeyboardDeviceBaseName
Type:          REG_SZ
Data:          KeyboardPort

```

```

Value 3
Name:          MouseDataQueueSize
Type:          REG_DWORD
Data:          0x64

```

```

Value 4
Name:          MouseResolution
Type:          REG_DWORD
Data:          0x3

```

```

Value 5
Name:          MouseSynchIn100ns
Type:          REG_DWORD
Data:          0x1312d00

```

```

Value 6
Name:          NumberOfButtons
Type:          REG_DWORD
Data:          0x2

```

```

Value 7
Name:          PointerDeviceBaseName
Type:          REG_SZ
Data:          PointerPort

```

```

Value 8
Name:          PollingIterations
Type:          REG_DWORD
Data:          0x2ee0

```

```

Value 9
Name:          PollingIterationsMaximum
Type:          REG_DWORD
Data:          0x2ee0

```

```

Value 10
Name:          PollStatusIterations
Type:          REG_DWORD
Data:          0x1

```

```

Value 11
Name:          ResendIterations
Type:          REG_DWORD
Data:          0x3

```

```

Value 12
Name:          SampleRate
Type:          REG_DWORD
Data:          0x3c

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\inetaccs
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 4:16 PM
Value 0
Name:          <NO NAME>
Type:          REG_SZ
Data:

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\inetaccs\Parameters
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 4:16 PM
Value 0
Name:          <NO NAME>
Type:          REG_SZ
Data:

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Inport
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          ErrorControl
Type:          REG_DWORD
Data:          0

```

```

Value 1
Name:          Group
Type:          REG_SZ
Data:          Pointer Port

```

```

Value 2
Name:          Start
Type:          REG_DWORD
Data:          0x4

```

```

Value 3
Name:          Tag
Type:          REG_DWORD
Data:          0x1

```

```

Value 4
Name:          Type
Type:          REG_DWORD
Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Inport\Parameters
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
Name:          HzMode
Type:          REG_DWORD
Data:          0x2

```

```

Value 1
Name:          MouseDataQueueSize
Type:          REG_DWORD
Data:          0x64

```

Value 2
 Name: NumberOfButtons
 Type: REG_DWORD
 Data: 0x2

Value 3
 Name: PointerDeviceBaseName
 Type: REG_SZ
 Data: PointerPort

Value 4
 Name: SampleRate
 Type: REG_DWORD
 Data: 0x32

Key Name:
 SYSTEM\CurrentControlSet\Services\Jazzg300
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Jazzg300\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: framebuffer

Value 1
 Name: Monitor
 Type: REG_SZ
 Data: drivers\jazz.bvp

Value 2
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\Jazzg364
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Jazzg364\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: framebuffer

Value 1
 Name: Monitor
 Type: REG_SZ
 Data: drivers\jazz.bvp

Value 2
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\Jzvxl484
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0

Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Jzvxl484\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: jzvxl484

Value 1
 Name: Monitor
 Type: REG_SZ
 Data: drivers\jazz.bvp

Value 2
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\Kbdclass
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 2:56 PM

Value 0
 Name: DisplayName
 Type: REG_SZ
 Data: Keyboard Class Driver

Value 1
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: Group
 Type: REG_SZ
 Data: Keyboard Class

Value 3
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: System32\DRIVERS\kbdclass.sys

Value 4
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 5
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Kbdclass\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_KBDCLASS\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Kbdclass\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ConnectMultiplePorts
 Type: REG_DWORD
 Data: 0

Value 1
 Name: KeyboardDataQueueSize
 Type: REG_DWORD
 Data: 0x64

Value 2
 Name: KeyboardDeviceBaseName
 Type: REG_SZ
 Data: KeyboardClass

Value 3
 Name: MaximumPortsServiced
 Type: REG_DWORD
 Data: 0x3

Key Name:
 SYSTEM\CurrentControlSet\Services\KSecDD
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ

Data: Base

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\KSecDD\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_KSECDD\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer
 Class Name: <NO CLASS>
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data: TDI

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ

Value 2
 Name: DisplayName
 Type: REG_SZ
 Data: Server

Value 3
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\services.exe

Value 5
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 6
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 7
 Name: Type
 Type: REG_DWORD
 Data: 0x20

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\AutotunedParameters
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_LANMANSERVER\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\Linkage
 Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: Bind
 Type: REG_MULTI_SZ
 Data: \Device\NetBT_N1001

Value 1
 Name: Export
 Type: REG_MULTI_SZ
 Data: \Device\LanmanServer\NetBT_N1001

Value 2
 Name: Route
 Type: REG_MULTI_SZ
 Data: "NetBT" "N100" "N1001"

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\Linkage\Disabled
 Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: Bind
 Type: REG_MULTI_SZ
 Data:

Value 1
 Name: Export
 Type: REG_MULTI_SZ
 Data:

Value 2
 Name: Route
 Type: REG_MULTI_SZ
 Data:

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters
 Class Name: GenericClass
 Last Write Time: 1/11/99 - 1:29 PM

Value 0
 Name: Lmannounce
 Type: REG_DWORD
 Data: 0

Value 1
 Name: MaxMpxCt
 Type: REG_DWORD
 Data: 0x1ff

Value 2
 Name: NullSessionPipes
 Type: REG_MULTI_SZ
 Data: COMNAP
 COMNODE
 SQL\QUERY
 SPOOLSS
 LLSRPC
 EPMAPPER
 LOCATOR

Value 3
 Name: NullSessionShares
 Type: REG_MULTI_SZ
 Data: COMCFG
 DFS\$

Value 4
 Name: Size
 Type: REG_DWORD
 Data: 0x3

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\Security
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: Security
 Type: REG_BINARY
 Data:
 00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 0
 00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
 00 18 00
 00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
 00 00 0
 00000050 74 00 42 00 00 00 1c 00 - fd 01 02 00 01
 02 00 0 t.B.....
 00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 54
 00 00 0#.T...
 00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
 00 00 05
 00000080 20 00 00 00 20 02 00 00 - 54 00 00 00 00
 00 1c 00 ...T.....
 00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
 00 00 0
 000000a0 25 02 00 00 54 00 00 00 - 00 00 18 00 fd
 01 02 00 %...T.....
 000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
 02 00 00%...
 000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
 01 00 00
 000000d0 00 00 00 05 12 00 00 00 -

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\Shares
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanServer\Shares\Security
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:13 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanWorkstation
 Class Name: <NO CLASS>
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ

Data: TDI

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data:

Value 2
 Name: DisplayName
 Type: REG_SZ
 Data: Workstation

Value 3
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Group
 Type: REG_SZ
 Data: NetworkProvider

Value 5
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\services.exe

Value 6
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 7
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 8
 Name: Type
 Type: REG_DWORD
 Data: 0x20

Key Name:
 SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_LANMANWORKSTATION\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

```

Key Name:
SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Linkage
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: Bind
Type: REG_MULTI_SZ
Data: \Device\NetBT_N1001

Value 1
Name: Export
Type: REG_MULTI_SZ
Data: \Device\LanmanWorkstation\NetBT_N1001

Value 2
Name: Route
Type: REG_MULTI_SZ
Data: "NetBT" "N100" "N1001"

Key Name:
SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Linkage\Disabled
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: Bind
Type: REG_MULTI_SZ
Data:

Value 1
Name: Export
Type: REG_MULTI_SZ
Data:

Value 2
Name: Route
Type: REG_MULTI_SZ
Data:

Key Name:
SYSTEM\CurrentControlSet\Services\LanmanWorkstation\NetworkProvider
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Devicename
Type: REG_SZ
Data: \Device\LanmanRedirector

Value 1
Name: Name
Type: REG_SZ
Data: Microsoft Windows Network

```

```

Value 2
Name: ProviderPath
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\ntlanman.dll

Key Name:
SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\LanmanWorkstation\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 01 08 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 cc
fa 14 00 .....#.
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - cc fa 14 00 00
00 1c 00 .....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 cc fa 14 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\LicenseInfo
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Start
Type: REG_DWORD

```

```

Data: 0x3

Value 2
Name: Type
Type: REG_DWORD
Data: 0x4

Key Name:
SYSTEM\CurrentControlSet\Services\LicenseInfo\FilePrint
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: ConcurrentLimit
Type: REG_DWORD
Data: 0

Value 1
Name: DisplayName
Type: REG_SZ
Data: Windows NT Server

Value 2
Name: FamilyDisplayName
Type: REG_SZ
Data: Windows NT Server

Value 3
Name: FlipAllow
Type: REG_DWORD
Data: 0

Value 4
Name: Mode
Type: REG_DWORD
Data: 0

Key Name:
SYSTEM\CurrentControlSet\Services\LicenseInfo\MSSQL7.0
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:28 PM
Value 0
Name: ConcurrentLimit
Type: REG_DWORD
Data: 0

Value 1
Name: DisplayName
Type: REG_SZ
Data: Microsoft SQL Server 7.0

Value 2
Name: FamilyDisplayName
Type: REG_SZ
Data: Microsoft SQL Server

Value 3
Name: FlipAllow
Type: REG_DWORD
Data: 0x1

```


Value 4
 Name: Mode
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService
 Class Name: <NO CLASS>
 Last Write Time: 1/19/99 - 5:27 PM

Value 0
 Name: DisplayName
 Type: REG_SZ
 Data: License Logging Service

Value 1
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\llssrv.exe

Value 3
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 4
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 5
 Name: Type
 Type: REG_DWORD
 Data: 0x10

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_LICENSESERVICE\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\File
 Print
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\File
 Print\KSecDD
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\File
 Print\MSAfpSrv
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\File
 Print\SMBServer
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\File
 Print\TCP/IP Print Server
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\Para
 meters
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Value 0
 Name: EnterpriseServer
 Type: REG_SZ
 Data:

Value 1
 Name: ReplicationTime
 Type: REG_DWORD
 Data: 0x15180

Value 2
 Name: ReplicationType
 Type: REG_DWORD
 Data: 0

Value 3
 Name: UseEnterprise
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\LicenseService\Secu
 rity
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM

Value 0
 Name: Security

Type: REG_BINARY
 Data:
 00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
 00 18 00
 00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000050 00 00 08 00 00 00 1c 00 - fd 01 02 00 01
 02 00 00
 00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 20
 01 14 00#...
 00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
 00 00 05
 00000080 20 00 00 00 20 02 00 00 - 20 01 14 00 00
 00 1c 00
 00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
 00 00 00
 000000a0 25 02 00 00 20 01 14 00 - 00 00 18 00 fd
 01 02 00 %...
 000000b0 01 01 00 00 00 00 05 05 - 12 00 00 00 25
 02 00 00%...
 000000c0 01 01 00 00 00 00 05 05 - 12 00 00 00 01
 01 00 00
 000000d0 00 00 00 05 12 00 00 00 -

Key Name:
 SYSTEM\CurrentControlSet\Services\LmHosts
 Class Name: <NO CLASS>
 Last Write Time: 1/19/99 - 5:28 PM

Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data: NetworkProvider

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data:

Value 2
 Name: DisplayName
 Type: REG_SZ
 Data: TCP/IP NetBIOS Helper

Value 3
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\services.exe

Value 5

Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 6
Name: Start
Type: REG_DWORD
Data: 0x3

Value 7
Name: Type
Type: REG_DWORD
Data: 0x20

Key Name:
SYSTEM\CurrentControlSet\Services\LmHosts\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_LMHOSTS\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\LmHosts\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\LmHosts\Linkage\Dis
abled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\LmHosts\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\LmHosts\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....

00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000050 00 00 00 01 00 00 1c 00 - fd 01 02 00 01
02 00 00
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 00
00 00 00#.....
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05
00000080 20 00 00 00 20 02 00 00 - 00 00 00 00 00
00 1c 00
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00
000000a0 25 02 00 00 00 00 00 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\Messenger
Class Name: <NO CLASS>
Last Write Time: 1/19/99 - 5:27 PM
Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data:

Value 1
Name: DependOnService
Type: REG_MULTI_SZ
Data: LanmanWorkstation
NetBios

Value 2
Name: DisplayName
Type: REG_SZ
Data: Messenger

Value 3
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 4
Name: ImagePath
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\services.exe

Value 5
Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 6

Name: Start
Type: REG_DWORD
Data: 0x3

Value 7
Name: Type
Type: REG_DWORD
Data: 0x20

Key Name:
SYSTEM\CurrentControlSet\Services\Messenger\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_MESSENGER\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Messenger\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Messenger\Linkage\D
isabled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Messenger\Parameter
s
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Messenger\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00

```

00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 05 00 00 1c 00 - fd 01 02 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 20
00 00 00 .....#...
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 20 00 00 00 00
00 1c 00 .....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 20 00 00 00 - 00 00 18 00 fd
01 02 00 %...
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mga
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 12:07 PM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: Group
Type: REG_SZ
Data: Video

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mga\Device0
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: InstalledDisplayDrivers
Type: REG_MULTI_SZ
Data: mga

```

```

Value 1
Name: VgaCompatible
Type: REG_DWORD
Data: 0

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mga_mil
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 12:07 PM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: Group
Type: REG_SZ
Data: Video

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mga_mil\Device0
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: InstalledDisplayDrivers
Type: REG_MULTI_SZ
Data: mga

```

```

Value 1
Name: VgaCompatible
Type: REG_DWORD
Data: 0

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mitsumi
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

```

```

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Tag
Type: REG_DWORD

```

```

Data: 0x1f
Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mkecr5xx
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

```

```

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x1d

```

```

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Modem
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: Group
Type: REG_SZ
Data: Extended base

```

```

Value 2
Name: PlugPlayServiceType
Type: REG_DWORD
Data: 0x2

```

```

Value 3
Name: Start
Type: REG_DWORD
Data: 0x3

```

```

Value 4

```

```

Name: Tag
Type: REG_DWORD
Data: 0x4

Value 5
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Modem\Parameters
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Mouclass
Class Name: <NO CLASS>
Last Write Time: 1/8/99 - 11:42 AM
Value 0
Name: DisplayName
Type: REG_SZ
Data: Mouse Class Driver

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 2
Name: Group
Type: REG_SZ
Data: Pointer Class

Value 3
Name: ImagePath
Type: REG_EXPAND_SZ
Data: System32\DRIVERS\mouclass.sys

Value 4
Name: Start
Type: REG_DWORD
Data: 0x1

Value 5
Name: Tag
Type: REG_DWORD
Data: 0x1

Value 6
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Mouclass\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_MOUCLASS\0000

```

```

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Mouclass\Parameters
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: ConnectMultiplePorts
Type: REG_DWORD
Data: 0x1

Value 1
Name: MaximumPortsServed
Type: REG_DWORD
Data: 0x3

Value 2
Name: MouseDataQueueSize
Type: REG_DWORD
Data: 0x64

Value 3
Name: PointerDeviceBaseName
Type: REG_SZ
Data: PointerClass

Key Name:
SYSTEM\CurrentControlSet\Services\MSDTC
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:34 PM
Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data:

Value 1
Name: DependOnService
Type: REG_MULTI_SZ
Data: RPCSS

Value 2
Name: DisplayName
Type: REG_SZ
Data: MSDTC

Value 3
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 4

```

```

Name: Group
Type: REG_SZ
Data: MS Transactions

Value 5
Name: ImagePath
Type: REG_EXPAND_SZ
Data: C:\WINNT40\System32\msdtc.exe

Value 6
Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 7
Name: Start
Type: REG_DWORD
Data: 0x3

Value 8
Name: Type
Type: REG_DWORD
Data: 0x10

Key Name:
SYSTEM\CurrentControlSet\Services\MSDTC\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM
Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 74 00 73 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 t.s.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 76
00 63 00 ....#.v.c.
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 76 00 63 00 00
00 1c 00 ...v.c....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 76 00 63 00 - 00 00 18 00 fd
01 02 00 %..v.c.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

Key Name:
 SYSTEM\CurrentControlSet\Services\Msfs
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: File system

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x2

Key Name:
 SYSTEM\CurrentControlSet\Services\Msfs\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_MSFS\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\MSSCNTRS
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\MSSCNTRS\Performance
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Value 0
 Name: Close
 Type: REG_SZ
 Data: Close

Value 1
 Name: Collect
 Type: REG_SZ

Data: Collect

Value 2
 Name: Library
 Type: REG_SZ
 Data: C:\PROGRA~1\COMMON~1\System\MSSearch\bin\msscncrs.dll

Value 3
 Name: Open
 Type: REG_SZ
 Data: Open

Key Name:
 SYSTEM\CurrentControlSet\Services\MSSEARCH
 Class Name: <NO CLASS>
 Last Write Time: 1/19/99 - 5:05 PM

Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data:

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data: RPCSS
 NTLMSSP

Value 2
 Name: DisplayName
 Type: REG_SZ
 Data: Microsoft Search

Value 3
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: C:\Program Files\Common Files\System\MSSearch\bin\mssearch.exe

Value 5
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 6
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 7
 Name: Type
 Type: REG_DWORD
 Data: 0x10

Key Name:
 SYSTEM\CurrentControlSet\Services\MSSEARCH\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_MSSEARCH\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\MSSEARCH\Performance
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:32 PM

Value 0
 Name: First Counter
 Type: REG_DWORD
 Data: 0x7a8

Value 1
 Name: First Help
 Type: REG_DWORD
 Data: 0x7a9

Value 2
 Name: Last Counter
 Type: REG_DWORD
 Data: 0x7ea

Value 3
 Name: Last Help
 Type: REG_DWORD
 Data: 0x7eb

Key Name:
 SYSTEM\CurrentControlSet\Services\MSSEARCH\Security
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Value 0
 Name: Security
 Type: REG_BINARY
 Data:
 00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
 00 18 00

```

00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 74 00 69 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 t.i.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 6f
00 6e 00 ....#.o.n.
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 6f 00 6e 00 00
00 1c 00 .....o.n.....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 6f 00 6e 00 - 00 00 18 00 fd
01 02 00 %...o.n.....
000000b0 01 01 00 00 00 00 05 00 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 05 00 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\MSSGatherer
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\MSSGatherer\Perform
ance
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM

```

```

Value 0
Name: First Counter
Type: REG_DWORD
Data: 0x738

```

```

Value 1
Name: First Help
Type: REG_DWORD
Data: 0x739

```

```

Value 2
Name: Last Counter
Type: REG_DWORD
Data: 0x78c

```

```

Value 3
Name: Last Help
Type: REG_DWORD
Data: 0x78d

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\MSSGTHRSVC
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:32 PM

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\MSSGTHRSVC\Performa
nce
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:32 PM

```

```

Value 0
Name: First Counter
Type: REG_DWORD
Data: 0x7ec

```

```

Value 1
Name: First Help
Type: REG_DWORD
Data: 0x7ed

```

```

Value 2
Name: Last Counter
Type: REG_DWORD
Data: 0x826

```

```

Value 3
Name: Last Help
Type: REG_DWORD
Data: 0x827

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mssindex
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\mssindex\Performanc
e
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:31 PM

```

```

Value 0
Name: First Counter
Type: REG_DWORD
Data: 0x78e

```

```

Value 1
Name: First Help
Type: REG_DWORD
Data: 0x78f

```

```

Value 2
Name: Last Counter
Type: REG_DWORD
Data: 0x7a6

```

```

Value 3
Name: Last Help
Type: REG_DWORD
Data: 0x7a7

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\MSSQLServer
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:30 PM

```

```

Value 0
Name: DisplayName
Type: REG_SZ
Data: MSSQLServer

```

```

Value 1
Name: ErrorControl

```

```

Type: REG_DWORD
Data: 0x1

```

```

Value 2
Name: ImagePath
Type: REG_EXPAND_SZ
Data: C:\MSSQL7\bin\sqlservr.exe

```

```

Value 3
Name: ObjectName
Type: REG_SZ
Data: .\Administrator

```

```

Value 4
Name: Start
Type: REG_DWORD
Data: 0x3

```

```

Value 5
Name: Type
Type: REG_DWORD
Data: 0x10

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\MSSQLServer\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

```

```

Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_MSSQLSERVER\0000

```

```

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

```

```

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\MSSQLServer\Performa
nce
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:34 PM

```

```

Value 0
Name: Close
Type: REG_SZ
Data: CloseSQLPerformanceData

```

```

Value 1
Name: Collect
Type: REG_SZ
Data: CollectSQLPerformanceData

```

```

Value 2
Name: First Counter
Type: REG_DWORD
Data: 0x828

```

Value 3
 Name: First Help
 Type: REG_DWORD
 Data: 0x829

Value 4
 Name: Last Counter
 Type: REG_DWORD
 Data: 0x92a

Value 5
 Name: Last Help
 Type: REG_DWORD
 Data: 0x92b

Value 6
 Name: Library
 Type: REG_SZ
 Data: SQLCTR70.DLL

Value 7
 Name: Open
 Type: REG_SZ
 Data: OpenSQLPerformanceData

Key Name:
 SYSTEM\CurrentControlSet\Services\MSSQLServer\Security

Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:32 PM

Value 0
 Name: Security
 Type: REG_BINARY
 Data:
 00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
 00 18 00
 00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000050 74 00 73 00 00 00 1c 00 - fd 01 02 00 01
 02 00 00 t.s.....
 00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 76
 00 63 00#.v.c.....
 00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
 00 00 05
 00000080 20 00 00 00 20 02 00 00 - 76 00 63 00 00
 00 1c 00 ...v.c.....
 00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
 00 00 00
 000000a0 25 02 00 00 76 00 63 00 - 00 00 18 00 fd
 01 02 00 %..v.c.....
 000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
 02 00 00%...
 000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
 01 00 00
 01 00 00

000000d0 00 00 00 05 12 00 00 00 -

Key Name:
 SYSTEM\CurrentControlSet\Services\Mup
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DisplayName
 Type: REG_SZ
 Data: Mup

Value 1
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: Group
 Type: REG_SZ
 Data: Network

Value 3
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: \SystemRoot\System32\drivers\mup.sys

Value 4
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 5
 Name: Type
 Type: REG_DWORD
 Data: 0x2

Key Name:
 SYSTEM\CurrentControlSet\Services\Mup\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_MUP\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Mup\Parameters
 Class Name: <NO CLASS>

Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\N100
 Class Name: <NO CLASS>
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: DisplayName
 Type: REG_SZ
 Data: Compaq Ethernet or Fast Ethernet
 NIC Driver

Value 1
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: Group
 Type: REG_SZ
 Data: NDIS

Value 3
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: \SystemRoot\System32\drivers\n100nt.sys

Value 4
 Name: Start
 Type: REG_DWORD
 Data: 0x2

Value 5
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\N100\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_N100\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\N100\Linkage
 Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:54 AM

```

Value 0
Name: Bind
Type: REG_MULTI_SZ
Data: \Device\N1001

Value 1
Name: Export
Type: REG_MULTI_SZ
Data: \Device\N1001

Value 2
Name: Route
Type: REG_MULTI_SZ
Data: "N1001"

Key Name:
SYSTEM\CurrentControlSet\Services\N100\Linkage\Disab
led
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: Bind
Type: REG_MULTI_SZ
Data:

Value 1
Name: Export
Type: REG_MULTI_SZ
Data:

Value 2
Name: Route
Type: REG_MULTI_SZ
Data:

Key Name:
SYSTEM\CurrentControlSet\Services\N100\Parameters
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM

Key Name:
SYSTEM\CurrentControlSet\Services\N100\Security
Class Name: <NO CLASS>
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....

```

```

00000050 6d 00 63 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 m.c.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 69
00 61 00 ....#.i.a.
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 69 00 61 00 00
00 1c 00 ...i.a....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 69 00 61 00 - 00 00 18 00 fd
01 02 00 %i.a.....
000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\N1001
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Start
Type: REG_DWORD
Data: 0x3

Value 2
Name: Type
Type: REG_DWORD
Data: 0x4

Key Name:
SYSTEM\CurrentControlSet\Services\N1001\Linkage
Class Name: GenericClass
Last Write Time: 2/5/99 - 6:34 PM
Value 0
Name: Bind
Type: REG_MULTI_SZ
Data: \Device\N1001

Value 1
Name: Export
Type: REG_MULTI_SZ
Data: \Device\N1001

Value 2
Name: Route
Type: REG_MULTI_SZ
Data: "N1001"

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\N1001\Linkage\Disab
led
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM

Key Name:
SYSTEM\CurrentControlSet\Services\N1001\Parameters
Class Name: GenericClass
Last Write Time: 2/8/99 - 4:10 PM
Value 0
Name: Adaptive_IFS
Type: REG_DWORD
Data: 0x1

Value 1
Name: BoardHasBridge
Type: REG_DWORD
Data: 0

Value 2
Name: BusNumber
Type: REG_DWORD
Data: 0x8

Value 3
Name: BusType
Type: REG_DWORD
Data: 0x5

Value 4
Name: BusTypeLocal
Type: REG_DWORD
Data: 0x5

Value 5
Name: Eid
Type: REG_DWORD
Data: 0x5fd76b74

Value 6
Name: ForceDpx
Type: REG_DWORD
Data: 0

Value 7
Name: IoBaseAddress
Type: REG_DWORD
Data: 0x9000

Value 8
Name: IRQ
Type: REG_DWORD
Data: 0x5

Value 9
Name: MapRegisters

```



```

Type:          REG_DWORD
Data:          0x40

Value 10
Name:          MediaType
Type:          REG_DWORD
Data:          0x1

Value 11
Name:          MemoryMappedBaseAddress
Type:          REG_DWORD
Data:          0xf79f0008

Value 12
Name:          MsPciScan
Type:          REG_DWORD
Data:          0x1

Value 13
Name:          NetworkAddress
Type:          REG_SZ
Data:          0

Value 14
Name:          NumCoalesce
Type:          REG_DWORD
Data:          0x8

Value 15
Name:          NumRfd
Type:          REG_DWORD
Data:          0x20

Value 16
Name:          NumTbdPerTcb
Type:          REG_DWORD
Data:          0xc

Value 17
Name:          NumTcb
Type:          REG_DWORD
Data:          0x10

Value 18
Name:          PerfOptims
Type:          REG_DWORD
Data:          0x2

Value 19
Name:          PhySlotNumber
Type:          REG_DWORD
Data:          0xa

Value 20
Name:          ProposeQNFTAddress
Type:          REG_SZ
Data:          00805FD76B74

Value 21
Name:          RxDmaCount
Type:          REG_DWORD
Data:          0

```

```

Value 22
Name:          RxFifo
Type:          REG_DWORD
Data:          0x8

Value 23
Name:          SlotNumber
Type:          REG_DWORD
Data:          0x4

Value 24
Name:          Speed
Type:          REG_DWORD
Data:          0

Value 25
Name:          Threshold
Type:          REG_DWORD
Data:          0x10

Value 26
Name:          TxDmaCount
Type:          REG_DWORD
Data:          0

Value 27
Name:          TxFifo
Type:          REG_DWORD
Data:          0x8

Value 28
Name:          Txmitwait
Type:          REG_DWORD
Data:          0x1

Value 29
Name:          UcodeSW
Type:          REG_DWORD
Data:          0x1

Value 30
Name:          UnderrunRetry
Type:          REG_DWORD
Data:          0x1

Value 31
Name:          UseManualPciAssign
Type:          REG_DWORD
Data:          0x1

Key Name:
SYSTEM\CurrentControlSet\Services\N1001\Parameters\Tc
pip
Class Name:    GenericClass
Last Write Time: 2/8/99 - 10:11 AM
Value 0
Name:          DefaultGateway
Type:          REG_MULTI_SZ
Data:

Value 1

```

```

Name:          EnableDHCP
Type:          REG_DWORD
Data:          0

Value 2
Name:          IPAddress
Type:          REG_MULTI_SZ
Data:          130.168.240.0

Value 3
Name:          IPInterfaceContext
Type:          REG_DWORD
Data:          0x1

Value 4
Name:          IPInterfaceContextMax
Type:          REG_DWORD
Data:          0x1

Value 5
Name:          LLInterface
Type:          REG_SZ
Data:

Value 6
Name:          PPTPFiltering
Type:          REG_DWORD
Data:          0

Value 7
Name:          RawIPAllowedProtocols
Type:          REG_MULTI_SZ
Data:          0

Value 8
Name:          SubnetMask
Type:          REG_MULTI_SZ
Data:          255.255.0.0

Value 9
Name:          TCPAllowedPorts
Type:          REG_MULTI_SZ
Data:          0

Value 10
Name:          UDPAllowedPorts
Type:          REG_MULTI_SZ
Data:          0

Value 11
Name:          UseZeroBroadcast
Type:          REG_DWORD
Data:          0

Key Name:
SYSTEM\CurrentControlSet\Services\Ncr53c9x
Class Name:    <NO CLASS>

```

Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x1

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ncr77c22
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 12:07 PM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 1
Name: Group
Type: REG_SZ
Data: Video

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ncr77c22\Device0
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: InstalledDisplayDrivers
Type: REG_MULTI_SZ
Data: vga
vga256

Value 1
Name: VgaCompatible
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Ncsc700
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x4

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Ncsc710
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x5

Value 4
Name: Type
Type: REG_DWORD

Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: DisplayName
Type: REG_SZ
Data: Microsoft NDIS System Driver

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 2
Name: Group
Type: REG_SZ
Data: NDIS

Value 3
Name: Start
Type: REG_DWORD
Data: 0x1

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_NDIS\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\MediaTypes
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\3C592
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x20596d50

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2
Name: token
Type: REG_SZ
Data: 3C592

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\3C597
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x70596d50

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2
Name: token
Type: REG_SZ
Data: 3C597

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\BONSAI
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x62110e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2

Name: token
Type: REG_SZ
Data: BONSAI

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\C320TNT
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x32530e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: C320TNT

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\DE425
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x5042a310

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2
Name: token
Type: REG_SZ
Data: DE425

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\DEC300
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x230a310

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2

Name: token
Type: REG_SZ
Data: DEC300

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\DEC422
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x2042a310

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2
Name: token
Type: REG_SZ
Data: DEC422

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\DURANGO
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x260110e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: DURANGO

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\ELNK3EISA
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x90506d50

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2

Name: token
Type: REG_SZ
Data: ELNK3EISA

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\ES3210

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x12949

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: ES3210

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\F70XX

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x6690e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: F70XX

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\FL32

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x1010d425

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xfffffffffff

Value 2

Name: token
Type: REG_SZ
Data: FL32

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\FLNK

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x776d50

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: FLNK

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\J2577A

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x4019f022

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2
Name: token
Type: REG_SZ
Data: J2577A

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\MAPLE

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x160110e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2

Name: token
Type: REG_SZ
Data: MAPLE

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\NE3200

Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x7cc3a

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: NE3200

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\NETFLEX3

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x20f1110e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2
Name: token
Type: REG_SZ
Data: NETFLEX3

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA
\NETFLEX3.1

Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x40f1110e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xf0ffffff

Value 2

Name: token
Type: REG_SZ
Data: NETFLEX3

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\NETFLX
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x61110e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: NETFLX

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\NF3500
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x84633a

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: NF3500

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\NPEISA.1
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: id
Type: REG_DWORD
Data: 0x2093a

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2

Name: token
Type: REG_SZ
Data: NPEISA

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\NPEISA.2
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: id
Type: REG_DWORD
Data: 0x3093a

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: NPEISA

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\P1990
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x604f42

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: P1990

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\RODAN
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x63110e

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2

Name: token
Type: REG_SZ
Data: RODAN

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\SKETHNT
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x2644d

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: SKETHNT

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\SKFENT
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x1644d

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2
Name: token
Type: REG_SZ
Data: SKFENT

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\SMC8232
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x80a34d

Value 1
Name: Mask
Type: REG_DWORD
Data: 0xffffffff

Value 2

Name: token
 Type: REG_SZ
 Data: SMC8232

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\TLNK3E
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x9c616d50

Value 1
 Name: Mask
 Type: REG_DWORD
 Data: 0xf0ffffff

Value 2
 Name: token
 Type: REG_SZ
 Data: TLNK3E

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\EISA\TLNK3EISA
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x90616d50

Value 1
 Name: Mask
 Type: REG_DWORD
 Data: 0xf0ffffff

Value 2
 Name: token
 Type: REG_SZ
 Data: TLNK3EISA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\AT1700
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x6413

Value 1

Name: token
 Type: REG_SZ
 Data: AT1700

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\EE16MC
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x628b

Value 1
 Name: token
 Type: REG_SZ
 Data: EE16MC

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\EINK527
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x41

Value 1
 Name: token
 Type: REG_SZ
 Data: EINK527

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\ELNK3MCA.1
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x627c

Value 1
 Name: token
 Type: REG_SZ
 Data: ELNK3MCA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\ELNK3MCA.2
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x627d

Value 1
 Name: token
 Type: REG_SZ
 Data: ELNK3MCA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\ELNK3MCA.3
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x61db

Value 1
 Name: token
 Type: REG_SZ
 Data: ELNK3MCA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\ELNK3MCA.4
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x62f6

Value 1
 Name: token
 Type: REG_SZ
 Data: ELNK3MCA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\ELNK3MCA.5
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x62f7

Value 1
 Name: token
 Type: REG_SZ
 Data: ELNK3MCA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\ELNKMC
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x6042

```

Value 1
Name: token
Type: REG_SZ
Data: ELNKMC

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\F30XX
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x70

Value 1
Name: token
Type: REG_SZ
Data: F30XX

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\HPMCA
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x63ca

Value 1
Name: token
Type: REG_SZ
Data: HPMCA

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\IBMENIIN
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0xffe0

Value 1
Name: token
Type: REG_SZ
Data: IBMENIIN

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\IBMTOKA
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: Id
Type: REG_DWORD

```

```

Data: 0xe000

Value 1
Name: token
Type: REG_SZ
Data: IBMTOKA

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\IBMTOKMC
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0xe001

Value 1
Name: token
Type: REG_SZ
Data: IBMTOKMC

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\IRMAtrac.1
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x5c1c

Value 1
Name: token
Type: REG_SZ
Data: IRMAtrac

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\IRMAtrac.2
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x5c1d

Value 1
Name: token
Type: REG_SZ
Data: IRMAtrac

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\NCR TOK
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id

```

```

Type: REG_DWORD
Data: 0x100

Value 1
Name: token
Type: REG_SZ
Data: NCR TOK

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\NPMCA
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x69

Value 1
Name: token
Type: REG_SZ
Data: NPMCA

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\OCTK16.1
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0xa84

Value 1
Name: token
Type: REG_SZ
Data: OCTK16

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\OCTK16.2
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0xa85

Value 1
Name: token
Type: REG_SZ
Data: OCTK16

Key Name:
SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\OCTK16.3
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0

```

Name: Id
Type: REG_DWORD
Data: 0xa86

Value 1
Name: token
Type: REG_SZ
Data: OCTK16

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\QUADENET.1
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x8f6d

Value 1
Name: token
Type: REG_SZ
Data: QUADENET

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\QUADENET.2
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x8f6a

Value 1
Name: token
Type: REG_SZ
Data: QUADENET

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\SKFMNT.1
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x83

Value 1
Name: token
Type: REG_SZ
Data: SKFMNT

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\SKFMNT.2
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0xab

Value 1
Name: token
Type: REG_SZ
Data: SKFMNT

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\STREAMER.1
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x8fa0

Value 1
Name: token
Type: REG_SZ
Data: STREAMER

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\STREAMER.2
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x8fa2

Value 1
Name: token
Type: REG_SZ
Data: STREAMER

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\STREAMER.3
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x8fa8

Value 1
Name: token
Type: REG_SZ
Data: STREAMER

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\STREAMER.4
Class Name: <NO CLASS>

Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x8faa

Value 1
Name: token
Type: REG_SZ
Data: STREAMER

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\TC\$4046E
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x51

Value 1
Name: token
Type: REG_SZ
Data: TC\$4046E

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\UBPS
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x7012

Value 1
Name: token
Type: REG_SZ
Data: UBPS

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\WAVELAN_MCA
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM

Value 0
Name: Id
Type: REG_DWORD
Data: 0x6a14

Value 1
Name: token
Type: REG_SZ
Data: WAVELAN_MCA

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\WD8003EA

Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x67c0

Value 1
 Name: token
 Type: REG_SZ
 Data: WD8003EA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\WD8003WA
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x67c2

Value 1
 Name: token
 Type: REG_SZ
 Data: WD8003WA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\WD8013EPA
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x61c8

Value 1
 Name: token
 Type: REG_SZ
 Data: WD8013EPA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\MCA\WD8013WPA
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x61c9

Value 1
 Name: token
 Type: REG_SZ
 Data: WD8013WPA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI

Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\3C590
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x590010b7

Value 1
 Name: token
 Type: REG_SZ
 Data: 3C590

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\3C595
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x595010b7

Value 1
 Name: token
 Type: REG_SZ
 Data: 3C595

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\3C905
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x905010b7

Value 1
 Name: token
 Type: REG_SZ
 Data: 3C905

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\ALANE0
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x59009004

Value 1
 Name: token

Type: REG_SZ
 Data: ALANE0

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\AMDPCI
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x2001022

Value 1
 Name: token
 Type: REG_SZ
 Data: AMDPCI

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\DC21040
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x21011

Value 1
 Name: token
 Type: REG_SZ
 Data: DC21040

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\DC21041
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x141011

Value 1
 Name: token
 Type: REG_SZ
 Data: DC21041

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\DC21140
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x91011

Value 1

Name: token
 Type: REG_SZ
 Data: DC21142

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 DC21142
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x191011

Value 1
 Name: token
 Type: REG_SZ
 Data: DC21142

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 DEFFA
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0xf1011

Value 1
 Name: token
 Type: REG_SZ
 Data: DEFFA

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 E100BPCI
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x12298086

Value 1
 Name: token
 Type: REG_SZ
 Data: E100BPCI

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 E10PCI
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x12268086

Value 1
 Name: token
 Type: REG_SZ
 Data: E10PCI

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 LEC
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x100110b6

Value 1
 Name: token
 Type: REG_SZ
 Data: LEC

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 NCPF
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0x111bc

Value 1
 Name: token
 Type: REG_SZ
 Data: NCPF

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 NETFLEX3.1
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0xf1300e11

Value 1
 Name: token
 Type: REG_SZ
 Data: NETFLEX3

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 NETFLEX3.2
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0xae320e11

Value 1
 Name: token
 Type: REG_SZ
 Data: NETFLEX3

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 NETFLEX3.3
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0xae340e11

Value 1
 Name: token
 Type: REG_SZ
 Data: NETFLEX3

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 NETFLEX3.4
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0xae350e11

Value 1
 Name: token
 Type: REG_SZ
 Data: NETFLEX3

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 NETFLEX3.5
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD
 Data: 0xae430e11

Value 1
 Name: token
 Type: REG_SZ
 Data: NETFLEX3

Key Name:
 SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\
 NETFLEX3.6
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:47 AM
 Value 0
 Name: Id
 Type: REG_DWORD

Data: 0xae400e11
Value 1
Name: token
Type: REG_SZ
Data: NETFLEX3

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\NETFLEX3.7
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0xf1500e11

Value 1
Name: token
Type: REG_SZ
Data: NETFLEX3

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\O100PCI
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x11108d

Value 1
Name: token
Type: REG_SZ
Data: O100PCI

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\OCE4XMP
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x13108d

Value 1
Name: token
Type: REG_SZ
Data: OCE4XMP

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\OCTK16
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id

Type: REG_DWORD
Data: 0x1108d

Value 1
Name: token
Type: REG_SZ
Data: OCTK16

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\RNSFDDI
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x22001112

Value 1
Name: token
Type: REG_SZ
Data: RNSFDDI

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\RTL8029
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x802910ec

Value 1
Name: token
Type: REG_SZ
Data: RTL8029

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\SKFPNT
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x40001148

Value 1
Name: token
Type: REG_SZ
Data: SKFPNT

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\SKTOKNT_PCI
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0

Name: Id
Type: REG_DWORD
Data: 0x42001148

Value 1
Name: token
Type: REG_SZ
Data: SKTOKNT_PCI

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\NetDetect\PCI\STREAMER
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 5:47 AM
Value 0
Name: Id
Type: REG_DWORD
Data: 0x181014

Value 1
Name: token
Type: REG_SZ
Data: STREAMER

Key Name: SYSTEM\CurrentControlSet\Services\NDIS\Parameters
Class Name: <NO CLASS>
Last Write Time: 1/28/99 - 7:02 PM
Value 0
Name: ProcessorAffinityMask
Type: REG_DWORD
Data: 0

Key Name: SYSTEM\CurrentControlSet\Services\NetBIOS
Class Name: <NO CLASS>
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data: TDI

Value 1
Name: DependOnService
Type: REG_MULTI_SZ
Data:

Value 2
Name: DisplayName
Type: REG_SZ
Data: NetBIOS Interface

Value 3
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 4
Name: Group

```

Type:          REG_SZ
Data:          NetBIOSGroup

Value 5
Name:          ImagePath
Type:          REG_EXPAND_SZ
Data:          \SystemRoot\System32\drivers\netbios.sys

Value 6
Name:          Start
Type:          REG_DWORD
Data:          0x3

Value 7
Name:          Type
Type:          REG_DWORD
Data:          0x2

Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOS\Enum
Class Name:    <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name:          0
Type:          REG_SZ
Data:          Root\LEGACY_NETBIOS\0000

Value 1
Name:          Count
Type:          REG_DWORD
Data:          0x1

Value 2
Name:          NextInstance
Type:          REG_DWORD
Data:          0x1

Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOS\Linkage
Class Name:    GenericClass
Last Write Time: 2/2/99 - 11:55 AM
Value 0
Name:          Bind
Type:          REG_MULTI_SZ
Data:          \Device\NetBT_N1001

Value 1
Name:          Export
Type:          REG_MULTI_SZ
Data:          \Device\Netbios\NetBT_N1001

Value 2
Name:          LanaMap
Type:          REG_BINARY
Data:          00000000 01 00
..

```

```

Value 3
Name:          Route
Type:          REG_MULTI_SZ
Data:          "NetBT" "N100" "N1001"

Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOS\Linkage\Dis
abled
Class Name:    GenericClass
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name:          Bind
Type:          REG_MULTI_SZ
Data:          Export
Name:          REG_MULTI_SZ
Data:          Value 1
Name:          Export
Type:          REG_MULTI_SZ
Data:          Value 2
Name:          Route
Type:          REG_MULTI_SZ
Data:          Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOS\Parameters
Class Name:    GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOS\Parameters\
Winsock
Class Name:    GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          HelperDllName
Type:          REG_EXPAND_SZ
Data:          %SystemRoot%\system32\wshnetbs.dll
Value 1
Name:          Mapping
Type:          REG_BINARY
Data:          00000000 02 00 00 00 03 00 00 00 - 11 00 00 00 05
00 00 00 .....
00000010 00 00 00 00 11 00 00 00 - 02 00 00 00 00
00 00 00 .....
Value 2
Name:          MaxSockAddrLength
Type:          REG_DWORD
Data:          0x14
Value 3
Name:          MinSockAddrLength
Type:          REG_DWORD
Data:          0x14

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOS\Security
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          Security
Type:          REG_BINARY
Data:          00000000 01 00 14 80 cc 00 00 00 - d8 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 98 00 - 06 00 00 00 00
03 18 00 .....
00000040 00 00 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
03 18 00 .....
00000070 ff 01 0f 00 01 01 00 00 - 00 00 00 05 12
00 00 00 .....
00000080 20 02 00 00 00 03 18 00 - 00 00 02 00 01
02 00 00 .....
00000090 00 00 00 05 20 00 00 00 - 23 02 00 00 00
03 18 00 .....#.....
000000a0 9d 00 00 00 01 01 00 00 - 00 00 00 05 04
00 00 00 .....
000000b0 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
02 00 00 #.....
000000c0 00 00 00 05 20 00 00 00 - 21 02 00 00 01
01 00 00 .....!.....
000000d0 00 00 00 05 12 00 00 00 - 01 01 00 00 00
00 00 05 .....
000000e0 12 00 00 00
....

Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOSInformation
Class Name:    GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          ErrorControl
Type:          REG_DWORD
Data:          0x1
Value 1
Name:          Start
Type:          REG_DWORD
Data:          0x3
Value 2
Name:          Type
Type:          REG_DWORD
Data:          0x4

Key Name:
SYSTEM\CurrentControlSet\Services\NetBIOSInformation\
Linkage

```

Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBIOSInformation\Linkage\Disabled
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBIOSInformation\Parameters
 Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:55 AM

Value 0
 Name: EnumExport1
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: LanaNum1
 Type: REG_DWORD
 Data: 0

Value 2
 Name: MaxLana
 Type: REG_DWORD
 Data: 0

Value 3
 Name: Route
 Type: REG_MULTI_SZ
 Data: "NetBT" "N100" "N1001"

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBT
 Class Name: <NO CLASS>
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data:

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data: Tcpip

Value 2
 Name: DisplayName
 Type: REG_SZ
 Data: WINS Client (TCP/IP)

Value 3
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Group

Type: REG_SZ
 Data: PNP_TDI

Value 5
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: \SystemRoot\System32\drivers\netbt.sys

Value 6
 Name: Start
 Type: REG_DWORD
 Data: 0x2

Value 7
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBT\Adapters
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBT\Adapters\N100
 1
 Class Name: GenericClass
 Last Write Time: 2/8/99 - 10:11 AM

Value 0
 Name: NameServer
 Type: REG_SZ
 Data:

Value 1
 Name: NameServerBackup
 Type: REG_SZ
 Data:

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBT\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_NETBT\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBT\Linkage

Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: Bind
 Type: REG_MULTI_SZ
 Data: \Device\N1001

Value 1
 Name: Export
 Type: REG_MULTI_SZ
 Data: \Device\NetBT_N1001

Value 2
 Name: OtherDependencies
 Type: REG_MULTI_SZ
 Data: Tcpip

Value 3
 Name: Route
 Type: REG_MULTI_SZ
 Data: "N100" "N1001"

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBT\Linkage\Disab
 led
 Class Name: GenericClass
 Last Write Time: 2/2/99 - 11:54 AM

Value 0
 Name: Bind
 Type: REG_MULTI_SZ
 Data:

Value 1
 Name: Export
 Type: REG_MULTI_SZ
 Data:

Value 2
 Name: Route
 Type: REG_MULTI_SZ
 Data:

Key Name:
 SYSTEM\CurrentControlSet\Services\NetBT\Parameters
 Class Name: GenericClass
 Last Write Time: 2/8/99 - 10:11 AM

Value 0
 Name: BcastNameQueryCount
 Type: REG_DWORD
 Data: 0x3

Value 1
 Name: BcastQueryTimeout
 Type: REG_DWORD
 Data: 0x2ee

Value 2

```

Name:          CacheTimeout
Type:          REG_DWORD
Data:          0x927c0

Value 3
Name:          EnabledDNS
Type:          REG_DWORD
Data:          0

Value 4
Name:          EnableLMHOSTS
Type:          REG_DWORD
Data:          0x1

Value 5
Name:          EnableProxy
Type:          REG_DWORD
Data:          0

Value 6
Name:          NameServerPort
Type:          REG_DWORD
Data:          0x89

Value 7
Name:          NameSrvQueryCount
Type:          REG_DWORD
Data:          0x3

Value 8
Name:          NameSrvQueryTimeout
Type:          REG_DWORD
Data:          0x5dc

Value 9
Name:          NbProvider
Type:          REG_SZ
Data:          _tcp

Value 10
Name:          ScopeID
Type:          REG_SZ
Data:

Value 11
Name:          SessionKeepAlive
Type:          REG_DWORD
Data:          0x493e0

Value 12
Name:          Size/Small/Medium/Large
Type:          REG_DWORD
Data:          0x1

Value 13
Name:          TransportBindName
Type:          REG_SZ
Data:          \Device\

Key Name:
SYSTEM\CurrentControlSet\Services\NetBT\Security
Class Name: <NO CLASS>

```

```

Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          Security
Type:          REG_BINARY
Data:          00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 79 00 73 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 y.s.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 00
00 0a 00 .....#.....
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 00 00 0a 00 00
00 1c 00 .....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 00 0a 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\NetDDE
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:06 AM
Value 0
Name:          DependOnService
Type:          REG_MULTI_SZ
Data:          NetDDESDM

Value 1
Name:          DisplayName
Type:          REG_SZ
Data:          Network DDE

Value 2
Name:          ErrorControl
Type:          REG_DWORD
Data:          0x1

Value 3
Name:          Group
Type:          REG_SZ
Data:          NetDDEGroup

Value 4
Name:          ImagePath
Type:          REG_EXPAND_SZ

```

```

Data:          %SystemRoot%\system32\netdde.exe

Value 5
Name:          ObjectName
Type:          REG_SZ
Data:          LocalSystem

Value 6
Name:          Start
Type:          REG_DWORD
Data:          0x3

Value 7
Name:          Type
Type:          REG_DWORD
Data:          0x20

Key Name:
SYSTEM\CurrentControlSet\Services\NetDDE\Security
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          Security
Type:          REG_BINARY
Data:          00000000 01 00 14 80 9c 00 00 00 - a8 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 68 00 - 04 00 00 00 00
03 18 00 .....h.....
00000040 8d 00 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
03 18 00 .....
00000070 8f 00 02 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
00000080 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
01 00 00 #.....
00000090 00 00 00 05 04 00 00 00 - 23 02 00 00 01
01 00 00 .....#.....
000000a0 00 00 00 05 12 00 00 00 - 01 01 00 00 00
00 00 05 .....
000000b0 12 00 00 00
....

Key Name:
SYSTEM\CurrentControlSet\Services\NetDDEsdm
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:06 AM
Value 0
Name:          DependOnService
Type:          REG_MULTI_SZ
Data:

Value 1

```

```

Name:      DisplayName
Type:      REG_SZ
Data:      Network DDE DSDM

Value 2
Name:      ErrorControl
Type:      REG_DWORD
Data:      0x1

Value 3
Name:      ImagePath
Type:      REG_EXPAND_SZ
Data:      %SystemRoot%\system32\netdde.exe

Value 4
Name:      ObjectName
Type:      REG_SZ
Data:      LocalSystem

Value 5
Name:      Start
Type:      REG_DWORD
Data:      0x3

Value 6
Name:      Type
Type:      REG_DWORD
Data:      0x20

Key Name:
SYSTEM\CurrentControlSet\Services\NetDDEdsdm\Security
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:      Security
Type:      REG_BINARY
Data:
00000000 01 00 14 80 9c 00 00 00 - a8 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 68 00 - 04 00 00 00 00
03 18 00 .....h.....
00000040 8d 00 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
03 18 00 .....
00000070 8f 00 02 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
00000080 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
01 00 00 #.....
00000090 00 00 00 05 04 00 00 00 - 23 02 00 00 01
01 00 00 .....#.
000000a0 00 00 00 05 12 00 00 00 - 01 01 00 00 00
00 00 05 .....
000000b0 12 00 00 00
....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\NetDetect
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:      DisplayName
Type:      REG_SZ
Data:      NetDetect

Value 1
Name:      ErrorControl
Type:      REG_DWORD
Data:      0x1

Value 2
Name:      ImagePath
Type:      REG_EXPAND_SZ
Data:      \SystemRoot\system32\drivers\netdct.sys

Value 3
Name:      Start
Type:      REG_DWORD
Data:      0x3

Value 4
Name:      Type
Type:      REG_DWORD
Data:      0x1

Key Name:
SYSTEM\CurrentControlSet\Services\NetDetect\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name:      0
Type:      REG_SZ
Data:      Root\LEGACY_NETDETECT\0000

Value 1
Name:      Count
Type:      REG_DWORD
Data:      0x1

Value 2
Name:      NextInstance
Type:      REG_DWORD
Data:      0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Netlogon
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:06 AM
Value 0
Name:      DependOnGroup
Type:      REG_MULTI_SZ
Data:

Value 1
Name:      DependOnService

```

```

Type:      REG_MULTI_SZ
Data:      LanmanWorkstation
LmHosts

Value 2
Name:      DisplayName
Type:      REG_SZ
Data:      Net Logon

Value 3
Name:      ErrorControl
Type:      REG_DWORD
Data:      0x1

Value 4
Name:      Group
Type:      REG_SZ
Data:      RemoteValidation

Value 5
Name:      ImagePath
Type:      REG_EXPAND_SZ
Data:      %SystemRoot%\System32\lsass.exe

Value 6
Name:      ObjectName
Type:      REG_SZ
Data:      LocalSystem

Value 7
Name:      Start
Type:      REG_DWORD
Data:      0x3

Value 8
Name:      Type
Type:      REG_DWORD
Data:      0x20

Key Name:
SYSTEM\CurrentControlSet\Services\Netlogon\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Netlogon\Linkage\Di
sabled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Netlogon\Parameters
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:      DBFlag
Type:      REG_SZ
Data:      0

Value 1
Name:      DisablePasswordChange

```

```

Type:          REG_DWORD
Data:          0

Value 2
Name:          Scripts
Type:          REG_EXPAND_SZ
Data:          %SystemRoot%\system32\repl\import\scripts

Value 3
Name:          Update
Type:          REG_SZ
Data:          no

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Netlogon\Security
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          Security
Type:          REG_BINARY
Data:          00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 72 00 76 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 r.v.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 69
00 63 00 ....#.i.c.
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 69 00 63 00 00
00 1c 00 ..i.c.
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 69 00 63 00 - 00 00 18 00 fd
01 02 00 %..i.c.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Npfs
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          ErrorControl
Type:          REG_DWORD
Data:          0x1

Value 1

```

```

Name:          Group
Type:          REG_SZ
Data:          File system

Value 2
Name:          Start
Type:          REG_DWORD
Data:          0x1

Value 3
Name:          Type
Type:          REG_DWORD
Data:          0x2

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Npfs\Aliases
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          lsass
Type:          REG_MULTI_SZ
Data:          netlogon
lsarpc
samr

```

```

Value 1
Name:          ntsvcs
Type:          REG_MULTI_SZ
Data:          srvsvc
wkssvc
eventlog
browser
msgsvc
svct1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Npfs\Enum
Class Name:    <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

```

```

Value 0
Name:          0
Type:          REG_SZ
Data:          Root\LEGACY_NPFS\0000

```

```

Value 1
Name:          Count
Type:          REG_DWORD
Data:          0x1

```

```

Value 2
Name:          NextInstance
Type:          REG_DWORD
Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Ntfs
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
Name:          ErrorControl
Type:          REG_DWORD
Data:          0x1

```

```

Value 1
Name:          Group
Type:          REG_SZ
Data:          File system

```

```

Value 2
Name:          Start
Type:          REG_DWORD
Data:          0x4

```

```

Value 3
Name:          Type
Type:          REG_DWORD
Data:          0x2

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Ntfs\Enum
Class Name:    <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

```

```

Value 0
Name:          0
Type:          REG_SZ
Data:          Root\LEGACY_NTFS\0000

```

```

Value 1
Name:          Count
Type:          REG_DWORD
Data:          0x1

```

```

Value 2
Name:          NextInstance
Type:          REG_DWORD
Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\NtLmSsp
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

```

```

Value 0
Name:          DisplayName
Type:          REG_SZ
Data:          NT LM Security Support Provider

```

```

Value 1
Name:          ErrorControl
Type:          REG_DWORD
Data:          0x1

```

```

Value 2
Name:          ImagePath
Type:          REG_EXPAND_SZ
Data:          %SystemRoot%\System32\SERVICES.EXE

```

```

Value 3
Name:          ObjectName
Type:          REG_SZ

```


Data: LocalSystem

Value 4
Name: Start
Type: REG_DWORD
Data: 0x3

Value 5
Name: Type
Type: REG_DWORD
Data: 0x20

Key Name:
SYSTEM\CurrentControlSet\Services\NtLmSsp\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_NTLMSSP\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\NtLmSsp\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\NtLmSsp\Linkage\Dis
abled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\NtLmSsp\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\NtLmSsp\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 cc 00 00 00 - d8 00 00 00 14
00 00 00
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00

00000030 20 02 00 00 02 00 98 00 - 06 00 00 00 00
03 18 00
00000040 00 00 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
02 00 00
00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
03 18 00
00000070 ff 01 0f 00 01 01 00 00 - 00 00 00 05 12
00 00 00
00000080 20 02 00 00 00 03 18 00 - 00 00 02 00 01
02 00 00
00000090 00 00 00 05 20 00 00 00 - 23 02 00 00 00
03 18 00
000000a0 9d 00 00 00 01 01 00 00 - 00 00 00 05 04
00 00 00
000000b0 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
02 00 00 #.....
000000c0 00 00 00 05 20 00 00 00 - 21 02 00 00 01
01 00 00
000000d0 00 00 00 05 12 00 00 00 - 01 01 00 00 00
00 00 05
000000e0 12 00 00 00
....

Key Name:
SYSTEM\CurrentControlSet\Services\Null
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: Base

Value 2
Name: Start
Type: REG_DWORD
Data: 0x1

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x1

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Null\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ

Data: Root\LEGACY_NULL\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Oliscsi
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

Value 3
Name: Tag
Type: REG_DWORD
Data: 0x3

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Parallel
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data: Parallel arbitrator

Value 1
Name: DependOnService
Type: REG_MULTI_SZ
Data: Parport

Value 2
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 3
 Name: Group
 Type: REG_SZ
 Data: Extended base

Value 4
 Name: Start
 Type: REG_DWORD
 Data: 0x2

Value 5
 Name: Tag
 Type: REG_DWORD
 Data: 0x2

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Parallel\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_PARALLEL\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Parallel\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Parport
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Parallel arbitrator

Value 2
 Name: Start

Type: REG_DWORD
 Data: 0x2

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Parport\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_PARPORT\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\ParVdm
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data: Parallel arbitrator

Value 1
 Name: DependOnService
 Type: REG_MULTI_SZ
 Data: Parport

Value 2
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 3
 Name: Group
 Type: REG_SZ
 Data: Extended base

Value 4
 Name: Start
 Type: REG_DWORD

Data: 0x2

Value 5
 Name: Tag
 Type: REG_DWORD
 Data: 0x2

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\ParVdm\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_PARVDM\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\ParVdm\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\PCIDump
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: PCI Configuration

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 4

Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\PCIDump\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:33 PM

Value 0
 Name: Count
 Type: REG_DWORD
 Data: 0

Value 1
 Name: NextInstance
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 5:42 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: System Bus Extender

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\3Co
 m Corporation
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\3Co
 m Corporation\3C589
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: elnk3

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnade3.inf

Value 3
 Name: Option
 Type: REG_SZ
 Data: ELNK3ISA509

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Acc
 ton
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Acc
 ton\EN2212
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadzz.inf

Value 3
 Name: Option
 Type: REG_SZ
 Data: ACCTONEN2216

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Acc
 ton\EN2216-PCMCIA-ETHERNET
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadzz.inf

Value 3
 Name: Option
 Type: REG_SZ
 Data: ACCTONEN2216

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Ada
 ptec, Inc.
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Ada
 ptec, Inc.\APA-1460 SCSI Host Adapter
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: SCSI

Value 1
 Name: Driver
 Type: REG_SZ
 Data: sparrow

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Com
 paq
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Com
 paq\Ethernet LAN Card
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 2
 Name: Driver
 Type: REG_SZ
 Data: cpqndis

Value 3
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadzz.inf

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\D-Link
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\D-Link\DE-650
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadzz.inf

Value 3
 Name: Option
 Type: REG_SZ
 Data: DLINKDE650

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Diehl ISDN
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Diehl ISDN\ISDN-DIVA\PCM
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ

Data: Diehl_DIVA_TED5ES

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Future Domain Corporation
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Future Domain Corporation\SCSI PCMCIA Credit Card Controller
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: SCSI

Value 1
 Name: Driver
 Type: REG_SZ
 Data: fd16_700

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\IBM
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\IBM\TOKEN RING
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: ibmtok

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadtk.inf

Value 3
 Name: Option
 Type: REG_SZ
 Data: IBMTOK

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\IBM Corp.
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\IBM Corp.\Ethernet
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: CardMemorySize
 Type: REG_DWORD
 Data: 0

Value 2
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 3
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 4
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadni.inf

Value 5
 Name: Option
 Type: REG_SZ
 Data: NE2000IBMCOMPAT

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Kingston Technology Corp.
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Kingston Technology Corp.\EtherX
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: CardMemorySize
 Type: REG_DWORD
 Data: 0

Value 2
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 3

Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 4
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadni.inf

Value 5
 Name: Option
 Type: REG_SZ
 Data: NE2000IBMCOMPAT

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\KME
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\KME
 \KXLC002
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: SCSI

Value 1
 Name: Driver
 Type: REG_SZ
 Data: qlmini

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\MAD
 GE
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\MAD
 GE\SMART 16/4 PCMCIA RINGNODE
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: madgemport

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadma.inf

Value 3

Name: Option
 Type: REG_SZ
 Data: MSMDGMPPCMCIA

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Mic
 rodyne
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Mic
 rodyne\NE4200
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: CardMemorySize
 Type: REG_DWORD
 Data: 0

Value 2
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 3
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 4
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadni.inf

Value 5
 Name: Option
 Type: REG_SZ
 Data: NE2000IBMCOMPAT

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Nat
 ional Semiconductor
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Nat
 ional Semiconductor\InfoMover NE4100
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: CardMemorySize
 Type: REG_DWORD
 Data: 0

Value 2
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 3
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 4
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadni.inf

Value 5
 Name: Option
 Type: REG_SZ
 Data: NE2000IBMCOMPAT

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\OLI
 COM
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\OLI
 COM\ETHERCOM
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: oce2xm

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadzz.inf

Value 3
 Name: Option
 Type: REG_SZ
 Data: GOCARD

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\OLI
 COM\TR 16/4 II
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: Driver
 Type: REG_SZ
 Data: octkl6

Value 2
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadzz.inf

Value 3
 Name: Option
 Type: REG_SZ
 Data: PCMCIA

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Socket Communications Inc
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Socket Communications Inc\Socket EA PCMCIA LAN Adapter Revision D
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0

Value 1
 Name: CardMemorySize
 Type: REG_DWORD
 Data: 0

Value 2
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 3
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 4
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadn2.inf

Value 5
 Name: Option
 Type: REG_SZ
 Data: NE2000SOCKETEA

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Socket Communications Inc\Socket EA PCMCIA LAN Adapter Revision E
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0

Value 1
 Name: CardMemorySize
 Type: REG_DWORD
 Data: 0

Value 2
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 3
 Name: Driver
 Type: REG_SZ
 Data: ne2000

Value 4
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadn2.inf

Value 5
 Name: Option
 Type: REG_SZ
 Data: NE2000SOCKETEA

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xircom\CreditCard 10Base-T
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xircom\CreditCard 10Base-T
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DeviceType
 Type: REG_SZ
 Data: NET

Value 1
 Name: InfFileName
 Type: REG_SZ
 Data: oemnadzz.inf

Value 2
 Name: Option
 Type: REG_SZ
 Data:

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xircom\CreditCard 10Base-T\5331
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: Driver
 Type: REG_SZ
 Data: ce2xps

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xircom\CreditCard 10Base-T\5FFA
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: Driver
 Type: REG_SZ
 Data: cendis3

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xircom\CreditCard 10Base-T\64BA
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: Driver
 Type: REG_SZ
 Data: ce2xps

Key Name:
 SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xircom\CreditCard 10Base-T\9C9B
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: AttributeMemorySize
 Type: REG_DWORD
 Data: 0x1000

Value 1
 Name: Driver
 Type: REG_SZ
 Data:

Data: cendis3

Key Name:
SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xir
com\CreditCard 10Base-T\9F5B
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: AttributeMemorySize
Type: REG_DWORD
Data: 0x1000

Value 1
Name: Driver
Type: REG_SZ
Data: ce2ndis3

Key Name:
SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xir
com\CreditCard 10Base-T\9FAB
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: AttributeMemorySize
Type: REG_DWORD
Data: 0x1000

Value 1
Name: Driver
Type: REG_SZ
Data: cendis3

Key Name:
SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xir
com\CreditCard 10Base-T\B00D
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: AttributeMemorySize
Type: REG_DWORD
Data: 0x1000

Value 1
Name: Driver
Type: REG_SZ
Data: cendis3

Key Name:
SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xir
com\CreditCard Ethernet+Modem 28.8
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: AttributeMemorySize
Type: REG_DWORD
Data: 0x1000

Value 1
Name: DeviceType

Type: REG_SZ
Data: NET

Value 2
Name: Driver
Type: REG_SZ
Data: cem28xps

Value 3
Name: InfFileName
Type: REG_SZ
Data: oemnadzz.inf

Key Name:
SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xir
com\CreditCard Ethernet+Modem II
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: AttributeMemorySize
Type: REG_DWORD
Data: 0x1000

Value 1
Name: DeviceType
Type: REG_SZ
Data: NET

Value 2
Name: Driver
Type: REG_SZ
Data: cm2ndis3

Value 3
Name: InfFileName
Type: REG_SZ
Data: oemnadzz.inf

Value 4
Name: Option
Type: REG_SZ
Data:

Key Name:
SYSTEM\CurrentControlSet\Services\Pcmcia\DataBase\Xir
com\CreditCard Token Ring
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: AttributeMemorySize
Type: REG_DWORD
Data: 0x1000

Value 1
Name: DeviceType
Type: REG_SZ
Data: NET

Value 2
Name: Driver
Type: REG_SZ

Data: ctndnt

Value 3
Name: InfFileName
Type: REG_SZ
Data: oemnadzz.inf

Value 4
Name: Option
Type: REG_SZ
Data:

Key Name:
SYSTEM\CurrentControlSet\Services\Pcmcia\Parameters
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Value 0
Name: Interrupt
Type: REG_DWORD
Data: 0

Key Name:
SYSTEM\CurrentControlSet\Services\PlugPlay
Class Name: <NO CLASS>
Last Write Time: 1/19/99 - 5:24 PM

Value 0
Name: DisplayName
Type: REG_SZ
Data: Plug and Play

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 2
Name: Group
Type: REG_SZ
Data: PlugPlay

Value 3
Name: ImagePath
Type: REG_EXPAND_SZ
Data: %SystemRoot%\system32\services.exe

Value 4
Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 5
Name: PlugPlayServiceType
Type: REG_DWORD
Data: 0x3

Value 6
Name: Start
Type: REG_DWORD
Data: 0x3

Value 7

```

Name: Type
Type: REG_DWORD
Data: 0x20

Key Name:
SYSTEM\CurrentControlSet\Services\pnpsia
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: DisplayName
Type: REG_SZ
Data: PnP ISA Enabler Driver

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0

Value 2
Name: Group
Type: REG_SZ
Data: Base

Value 3
Name: Start
Type: REG_DWORD
Data: 0x1

Value 4
Name: Tag
Type: REG_DWORD
Data: 0xe

Value 5
Name: Type
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\pnpsia\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: Count
Type: REG_DWORD
Data: 0

Value 1
Name: NextInstance
Type: REG_DWORD
Data: 0

Key Name:
SYSTEM\CurrentControlSet\Services\ProtectedStorage
Class Name: <NO CLASS>
Last Write Time: 1/19/99 - 5:05 PM
Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data:

```

```

Value 1
Name: DependOnService
Type: REG_MULTI_SZ
Data: RpcSs

Value 2
Name: DisplayName
Type: REG_SZ
Data: Protected Storage

Value 3
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 4
Name: ImagePath
Type: REG_EXPAND_SZ
Data: c:\winnt40\system32\pstores.exe

Value 5
Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 6
Name: Start
Type: REG_DWORD
Data: 0x3

Value 7
Name: Type
Type: REG_DWORD
Data: 0x110

Key Name:
SYSTEM\CurrentControlSet\Services\ProtectedStorage\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: Count
Type: REG_SZ
Data: Root\LEGACY_PROTECTEDSTORAGE\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ProtectedStorage\Parameters
Class Name: <NO CLASS>

```

```

Last Write Time: 12/11/98 - 3:41 PM
Value 0
Name: AuthCodeCfg
Type: REG_DWORD
Data: 0x1

Value 1
Name: Configuration
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ProtectedStorage\Parameters\S2
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:22 PM
Value 0
Name: psbase.dll
Type: REG_BINARY
Data:
00000000 e6 ba 57 e2 1b 88 fc 20 - 01 ed 35 32 7e
8f 36 97 ..W.... .52~.6.
00000010 41 4c e1 76 97 ef 60 09 - 38 ea 86 6b 43
d6 30 4e AL.v..`.8..kC.ON
00000020 10 6b c1 75 61 84 d2 1b - c7 eb b2 18 f1
13 cb a0 .k.ua.....
00000030 6c 8f 31 65 8e 00 93 5e - 72 78 97 61 63
d1 74 9f 1.le..^rx.ac.t.
00000040 19 e6 b4 08 fc ba ba df - b1 9e da 17 19
5d 95 22 .....].
00000050 f9 f6 60 40 ae c4 72 0b - d1 74 7c 86 a8
8a f8 d2 ..`.r..tl....
00000060 ba 14 63 f2 a5 25 b9 a2 - b7 86 58 1e 4e
92 28 cf .c.%.X.N.(.
00000070 41 6a b7 57 4c 44 67 51 - 48 47 85 f5 6e
00 99 af Aj.WLDgQHg.n...
00000080 00 00 00 00 00 00 00 00 -
.....

Value 1
Name: pstores.exe
Type: REG_BINARY
Data:
00000000 85 88 4d da 80 23 97 51 - 7d ed 3f 28 64
e7 34 cc .M.#.Q)?.(d.4.
00000010 1b ff d9 8e 95 1b 60 d4 - 72 df 64 a3 e2
a3 7d af .....r.d..}.
00000020 4b 1a f3 19 cb f3 6d f2 - 3d bb 30 b6 fe
35 fe 66 K.....m=.0..5.f
00000030 1e 29 f5 a0 bc d9 b3 07 - 9e 02 b9 ac f2
37 70 8b .).....7p.
00000040 4f 87 41 b8 a9 66 ab cf - 4d 1e 54 50 68
d0 51 a0 O.A..f..M.TPh.Q.
00000050 48 af 44 1c 23 b7 27 f8 - 4a 7b 86 54 5a
42 86 e2 H.D.#.'J(.TZB..
00000060 a3 c2 07 0f ef 13 a2 72 - b4 8a e7 31 d4
4a d3 ef .....r...1.J..
00000070 3a 11 9d 51 2f 52 ec c6 - 6c d3 28 43 45
30 e2 72 ..Q/R..1.(CEO.r
00000080 00 00 00 00 00 00 00 00 -
.....

```



```

Key Name:
SYSTEM\CurrentControlSet\Services\ProtectedStorage\Se
curity
Class Name:      <NO CLASS>
Last Write Time: 12/11/98 - 3:41 PM
Value 0
  Name:          Security
  Type:          REG_BINARY
  Data:
00000000  01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010  34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020  ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030  20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040  8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050  f0 21 fa 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 .!.....
00000060  00 00 00 05 20 00 00 00 - 23 02 00 00 00
00 00 00 .....#.....
00000070  00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080  20 00 00 00 20 02 00 00 - 00 00 00 00 00
00 1c 00 .....
00000090  ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0  25 02 00 00 00 00 00 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0  01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0  01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0  00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\psidisp
Class Name:      <NO CLASS>
Last Write Time: 12/11/98 - 12:07 PM
Value 0
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0
Value 1
  Name:          Group
  Type:          REG_SZ
  Data:          Video
Value 2
  Name:          Start
  Type:          REG_DWORD
  Data:          0x4
Value 3
  Name:          Type
  Type:          REG_DWORD
  Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\psidisp\Device0
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:          InstalledDisplayDrivers
  Type:          REG_MULTI_SZ
  Data:          psidisp

```

```

Value 1
  Name:          VgaCompatible
  Type:          REG_DWORD
  Data:          0

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Q110wnt
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0x1

```

```

Value 1
  Name:          Group
  Type:          REG_SZ
  Data:          SCSI miniport

```

```

Value 2
  Name:          Start
  Type:          REG_DWORD
  Data:          0x4

```

```

Value 3
  Name:          Tag
  Type:          REG_DWORD
  Data:          0x23

```

```

Value 4
  Name:          Type
  Type:          REG_DWORD
  Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\qv
Class Name:      <NO CLASS>
Last Write Time: 12/11/98 - 12:07 PM

```

```

Value 0
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0

```

```

Value 1
  Name:          Group
  Type:          REG_SZ
  Data:          Video

```

```

Value 2

```

```

Name:          Start
Type:          REG_DWORD
Data:          0x4

```

```

Value 3
  Name:          Type
  Type:          REG_DWORD
  Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\qv\Device0
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Value 0
  Name:          InstalledDisplayDrivers
  Type:          REG_MULTI_SZ
  Data:          vga
          qv

```

```

Value 1
  Name:          VgaCompatible
  Type:          REG_DWORD
  Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Rdr
Class Name:      <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

```

```

Value 0
  Name:          DisplayName
  Type:          REG_SZ
  Data:          Rdr

```

```

Value 1
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0x1

```

```

Value 2
  Name:          Group
  Type:          REG_SZ
  Data:          Network

```

```

Value 3
  Name:          ImagePath
  Type:          REG_EXPAND_SZ
  Data:          \SystemRoot\System32\drivers\rdr.sys

```

```

Value 4
  Name:          Start
  Type:          REG_DWORD
  Data:          0x3

```

```

Value 5
  Name:          Type
  Type:          REG_DWORD
  Data:          0x2

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Rdr\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
  Name: 0
  Type: REG_SZ
  Data: Root\LEGACY_RDR\0000

Value 1
  Name: Count
  Type: REG_DWORD
  Data: 0x1

Value 2
  Name: NextInstance
  Type: REG_DWORD
  Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Rdr\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Rdr\Linkage\Disabled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Rdr\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Rdr\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
  Name: Security
  Type: REG_BINARY
  Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 02
00 04 00 .....#.....
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 02 00 04 00 00
00 1c 00 .....

```

```

00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 02 00 04 00 - 00 00 18 00 fd
01 02 00 %.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\Replicator
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
  Name: DependOnGroup
  Type: REG_MULTI_SZ
  Data:

Value 1
  Name: DependOnService
  Type: REG_MULTI_SZ
  Data: LanmanWorkstation
LanmanServer

Value 2
  Name: DisplayName
  Type: REG_SZ
  Data: Directory Replicator

Value 3
  Name: ErrorControl
  Type: REG_DWORD
  Data: 0x1

Value 4
  Name: ImagePath
  Type: REG_EXPAND_SZ
  Data: %SystemRoot%\System32\lmrepl.exe

Value 5
  Name: ObjectName
  Type: REG_SZ
  Data: LocalSystem

Value 6
  Name: Start
  Type: REG_DWORD
  Data: 0x3

Value 7
  Name: Type
  Type: REG_DWORD
  Data: 0x10

Key Name:
SYSTEM\CurrentControlSet\Services\Replicator\Exports
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Replicator\Imports
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Replicator\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Replicator\Linkage\Disabled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Replicator\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
  Name: ExportPath
  Type: REG_EXPAND_SZ
  Data: %SystemRoot%\System32\Repl\Export

Value 1
  Name: ImportPath
  Type: REG_EXPAND_SZ
  Data: %SystemRoot%\System32\Repl\Import

Value 2
  Name: Replicate
  Type: REG_DWORD
  Data: 0x2

Key Name:
SYSTEM\CurrentControlSet\Services\Replicator\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
  Name: Security
  Type: REG_BINARY
  Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 02
00 04 00 .....#.....
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 02 00 04 00 00
00 00 05 .....

```

```

00000080 20 00 00 00 20 02 00 00 - 02 00 04 00 00
00 1c 00  ....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00  ....
000000a0 25 02 00 00 02 00 04 00 - 00 00 18 00 fd
01 02 00  %.....
000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
02 00 00  .....%...
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
01 00 00  .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RPCLOCATOR
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 3:41 PM
Value 0
Name: DependOnService
Type: REG_MULTI_SZ
Data: LanmanWorkstation
Rdr

Value 1
Name: DisplayName
Type: REG_SZ
Data: Remote Procedure Call (RPC)
Locator

Value 2
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 3
Name: ImagePath
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\LOCATOR.EXE

Value 4
Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 5
Name: Start
Type: REG_DWORD
Data: 0x3

Value 6
Name: Type
Type: REG_DWORD
Data: 0x10

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RPCLOCATOR\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RPCLOCATOR\Linkage\
Disabled
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RPCLOCATOR\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: ExpirationAge
Type: REG_DWORD
Data: 0xe10

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RPCLOCATOR\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

```

```

Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 cc 00 00 00 - d8 00 00 00 14
00 00 00  ....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00  4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00  .....
00000030 20 02 00 00 02 00 98 00 - 06 00 00 00 00
03 18 00  .....
00000040 00 00 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00  .....
00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
02 00 00  .....
00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
03 18 00  .....
00000070 ff 01 0f 00 01 01 00 00 - 00 00 00 05 12
00 00 00  .....
00000080 20 02 00 00 00 03 18 00 - 00 00 02 00 01
02 00 00  .....
00000090 00 00 00 05 20 00 00 00 - 23 02 00 00 00
03 18 00  ...#.....
000000a0 9d 00 00 00 01 01 00 00 - 00 00 00 05 04
00 00 00  .....
000000b0 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
02 00 00  #.....
000000c0 00 00 00 05 20 00 00 00 - 21 02 00 00 01
01 00 00  .....
000000d0 00 00 00 05 12 00 00 00 - 01 01 00 00 00
00 00 05  .....
000000e0 12 00 00 00
....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RpcSs
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: DisplayName

```

```

Type: REG_SZ
Data: Remote Procedure Call (RPC)
Service

Value 1
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 2
Name: ImagePath
Type: REG_EXPAND_SZ
Data: %SystemRoot%\system32\RpcSs.exe

Value 3
Name: ObjectName
Type: REG_SZ
Data: LocalSystem

Value 4
Name: Start
Type: REG_DWORD
Data: 0x2

Value 5
Name: Type
Type: REG_DWORD
Data: 0x10

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RpcSs\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_RPCSS\0000

```

```

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RpcSs\Parameters
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\RpcSs\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Security
Type: REG_BINARY
Data:

```

```

00000000 01 00 14 80 cc 00 00 00 - d8 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 98 00 - 06 00 00 00 00
03 18 00 .....
00000040 00 00 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
03 18 00 .....
00000070 ff 01 0f 00 01 01 00 00 - 00 00 00 05 12
00 00 00 .....
00000080 20 02 00 00 00 03 18 00 - 00 00 02 00 01
02 00 00 .....
00000090 00 00 00 05 20 00 00 00 - 23 02 00 00 00
03 18 00 .....#.....
000000a0 9d 00 00 00 01 01 00 00 - 00 00 00 05 04
00 00 00 .....
000000b0 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
02 00 00 .....#.....
000000c0 00 00 00 05 20 00 00 00 - 21 02 00 00 01
01 00 00 .....!.!.....
000000d0 00 00 00 05 12 00 00 00 - 01 01 00 00 00
00 00 05 .....
000000e0 12 00 00 00
....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\s3
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 12:07 PM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: Group
Type: REG_SZ
Data: Video

```

```

Value 2
Name: Start
Type: REG_DWORD
Data: 0x4

```

```

Value 3
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\s3\Device0
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name: InstalledDisplayDrivers

```

```

Type: REG_MULTI_SZ
Data: s3

```

```

Value 1
Name: VgaCompatible
Type: REG_DWORD
Data: 0

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\SCache
Class Name: REG_SZ
Last Write Time: 1/21/99 - 3:46 PM
Value 0

```

```

Name: ErrorControl
Type: REG_DWORD
Data: 0x1

```

```

Value 1
Name: Group
Type: REG_SZ
Data: SuperDisk

```

```

Value 2
Name: KeyCode
Type: REG_SZ
Data: B5J2E18-199-159-150-135-166X

```

```

Value 3
Name: Start
Type: REG_DWORD
Data: 0

```

```

Value 4
Name: Type
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\SCache\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0

```

```

Name: Count
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: NextInstance
Type: REG_DWORD
Data: 0

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\SCache\Parameters
Class Name: REG_SZ
Last Write Time: 1/21/99 - 3:46 PM
Value 0

```

```

Name: BreakOnEntry
Type: REG_DWORD
Data: 0

```

```

Value 1
Name: DebugLevel
Type: REG_DWORD
Data: 0

```

```

Value 2
Name: devChar
Type: REG_DWORD
Data: 0x911

```

```

Value 3
Name: DosDevice
Type: REG_SZ
Data: C:

```

```

Value 4
Name: Drive
Type: REG_DWORD
Data: 0x17

```

```

Value 5
Name: LazyWrite
Type: REG_DWORD
Data: 0

```

```

Value 6
Name: MaxLRU
Type: REG_DWORD
Data: 0x2000

```

```

Value 7
Name: MinLRU
Type: REG_DWORD
Data: 0x20

```

```

Value 8
Name: NpParm
Type: REG_DWORD
Data: 0x32e18

```

```

Value 9
Name: PageSize
Type: REG_DWORD
Data: 0x8000

```

```

Value 10
Name: Partition
Type: REG_DWORD
Data: 0x2

```

```

Value 11
Name: PendingLimit
Type: REG_DWORD
Data: 0x10

```

```

Value 12
Name: SystemDevice
Type: REG_DWORD
Data: 0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\ScAlloc
Class Name: <NO CLASS>
Last Write Time: 1/21/99 - 3:55 PM
Value 0
  Name:      DisplayName
  Type:      REG_SZ
  Data:      ScAlloc

Value 1
  Name:      ErrorControl
  Type:      REG_DWORD
  Data:      0x1

Value 2
  Name:      ImagePath
  Type:      REG_EXPAND_SZ
  Data:      %SystemRoot%\System32\ScAllSrv.exe

Value 3
  Name:      ObjectName
  Type:      REG_SZ
  Data:      LocalSystem

Value 4
  Name:      Start
  Type:      REG_DWORD
  Data:      0x3

Value 5
  Name:      Type
  Type:      REG_DWORD
  Data:      0x110

Key Name:
SYSTEM\CurrentControlSet\Services\ScAlloc\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
  Name:      0
  Type:      REG_SZ
  Data:      Root\LEGACY_SCALLOC\0000

Value 1
  Name:      Count
  Type:      REG_DWORD
  Data:      0x1

Value 2
  Name:      NextInstance
  Type:      REG_DWORD
  Data:      0x1

Key Name:
SYSTEM\CurrentControlSet\Services\ScAlloc\Security
Class Name: <NO CLASS>
Last Write Time: 1/21/99 - 3:46 PM
Value 0
  Name:      Security
  Type:      REG_BINARY
  Data:

```

```

00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 74 00 6f 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 t.o.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 72
00 00 00 ....#.r...
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 72 00 00 00 00
00 1c 00 ...r.....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 72 00 00 00 - 00 00 18 00 fd
01 02 00 %..r.....
000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\Schedule
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:      ErrorControl
  Type:      REG_DWORD
  Data:      0x1

Value 1
  Name:      ImagePath
  Type:      REG_EXPAND_SZ
  Data:      %SystemRoot%\System32\AtSvc.Exe

Value 2
  Name:      ObjectName
  Type:      REG_SZ
  Data:      LocalSystem

Value 3
  Name:      Start
  Type:      REG_DWORD
  Data:      0x3

Value 4
  Name:      Type
  Type:      REG_DWORD
  Data:      0x10

Key Name:
SYSTEM\CurrentControlSet\Services\Scsiprnt
Class Name: <NO CLASS>

```

```

Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:      DependOnGroup
  Type:      REG_MULTI_SZ
  Data:      SCSI miniport

Value 1
  Name:      ErrorControl
  Type:      REG_DWORD
  Data:      0

Value 2
  Name:      Group
  Type:      REG_SZ
  Data:      Extended base

Value 3
  Name:      Start
  Type:      REG_DWORD
  Data:      0x2

Value 4
  Name:      Tag
  Type:      REG_DWORD
  Data:      0x3

Value 5
  Name:      Type
  Type:      REG_DWORD
  Data:      0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Scsiprnt\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:34 PM
Value 0
  Name:      Count
  Type:      REG_DWORD
  Data:      0

Value 1
  Name:      NextInstance
  Type:      REG_DWORD
  Data:      0

Key Name:
SYSTEM\CurrentControlSet\Services\Scsiscan
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:      DependOnGroup
  Type:      REG_MULTI_SZ
  Data:      SCSI miniport

Value 1
  Name:      ErrorControl
  Type:      REG_DWORD
  Data:      0

```

Value 2
 Name: Group
 Type: REG_SZ
 Data: SCSI Class

Value 3
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Tag
 Type: REG_DWORD
 Data: 0x3

Value 5
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Scsiscan\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: Count
 Type: REG_DWORD
 Data: 0

Value 1
 Name: NextInstance
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\Serial
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 2:55 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: ForceFifoEnable
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: Group
 Type: REG_SZ
 Data: Extended base

Value 3
 Name: LogFifo
 Type: REG_DWORD
 Data: 0

Value 4
 Name: PermitShare
 Type: REG_DWORD

Data: 0

Value 5
 Name: RxFIFO
 Type: REG_DWORD
 Data: 0x8

Value 6
 Name: ShareInterrupt
 Type: REG_DWORD
 Data: 0

Value 7
 Name: Start
 Type: REG_DWORD
 Data: 0x2

Value 8
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 9
 Name: TxFIFO
 Type: REG_DWORD
 Data: 0x1

Value 10
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Serial\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_SERIAL\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Serial\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Sermouse
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0

Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Pointer Port

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x2

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Sermouse\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: MouseDataQueueSize
 Type: REG_DWORD
 Data: 0x64

Value 1
 Name: NumberOfButtons
 Type: REG_DWORD
 Data: 0x2

Value 2
 Name: PointerDeviceBaseName
 Type: REG_SZ
 Data: PointerPort

Value 3
 Name: SampleRate
 Type: REG_DWORD
 Data: 0x28

Key Name:
 SYSTEM\CurrentControlSet\Services\Sfloppy
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DependOnGroup
 Type: REG_MULTI_SZ
 Data: SCSI miniport

Value 1
 Name: ErrorControl
 Type: REG_DWORD

Data: 0

Value 2
 Name: Group
 Type: REG_SZ
 Data: Primary disk

Value 3
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Tag
 Type: REG_DWORD
 Data: 0x4

Value 5
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Sfloppy\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: Count
 Type: REG_DWORD
 Data: 0

Value 1
 Name: NextInstance
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\Simbad
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: Filter

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 4

Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\slcd32
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x1c

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Sparrow
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x7

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Spock
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x2

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Spooler
 Class Name: <NO CLASS>
 Last Write Time: 1/19/99 - 5:27 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SpoolerGroup

Value 2
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\system32\spoolss.exe

Value 3
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 4
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 5
 Name: Type

Type: REG_DWORD
Data: 0x110

Key Name:
SYSTEM\CurrentControlSet\Services\Spooler\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_SPOOLER\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Spooler\Parameters
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\SQLServerAgent
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:34 PM

Value 0
Name: DependOnGroup
Type: REG_MULTI_SZ
Data:

Value 1
Name: DependOnService
Type: REG_MULTI_SZ
Data: MSSQLServer

Value 2
Name: DisplayName
Type: REG_SZ
Data: SQLServerAgent

Value 3
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 4
Name: ImagePath
Type: REG_EXPAND_SZ
Data: C:\MSSQL7\bin\sqlagent.exe

Value 5
Name: ObjectName
Type: REG_SZ
Data: .\Administrator

Value 6
Name: Start
Type: REG_DWORD
Data: 0x3

Value 7
Name: Type
Type: REG_DWORD
Data: 0x10

Key Name:
SYSTEM\CurrentControlSet\Services\SQLServerAgent\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:34 PM

Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00
00000050 74 00 73 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 t.s.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 76
00 63 00 # . . v . c .
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05
00000080 20 00 00 00 20 02 00 00 - 76 00 63 00 00
00 1c 00 v . c
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00
000000a0 25 02 00 00 76 00 63 00 - 00 00 18 00 fd
01 02 00 % . . v . c
000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
02 00 00 %
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
01 00 00
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\Srv
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

Value 0
Name: DisplayName
Type: REG_SZ
Data: Srv

Value 1
Name: ErrorControl
Type: REG_DWORD

Data: 0x1

Value 2
Name: Group
Type: REG_SZ
Data: Network

Value 3
Name: ImagePath
Type: REG_EXPAND_SZ
Data: \SystemRoot\System32\drivers\srv.sys

Value 4
Name: Start
Type: REG_DWORD
Data: 0x3

Value 5
Name: Type
Type: REG_DWORD
Data: 0x2

Key Name:
SYSTEM\CurrentControlSet\Services\Srv\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM

Value 0
Name: 0
Type: REG_SZ
Data: Root\LEGACY_SRV\0000

Value 1
Name: Count
Type: REG_DWORD
Data: 0x1

Value 2
Name: NextInstance
Type: REG_DWORD
Data: 0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Srv\Linkage
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Srv\Linkage\Disable
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Srv\Parameters
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Srv\Security
Class Name: <NO CLASS>


```

Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 76 00 69 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 v.i.....
00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 63
00 65 00 .....#...c.e.
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 63 00 65 00 00
00 1c 00 ...c.e....
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 63 00 65 00 - 00 00 18 00 fd
01 02 00 %...c.e.....
000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\symc810
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: ErrorControl
Type: REG_DWORD
Data: 0x1

Value 1
Name: Group
Type: REG_SZ
Data: SCSI miniport

Value 2
Name: ImagePath
Type: REG_EXPAND_SZ
Data: System32\DRIVERS\symc810.sys

Value 3
Name: PlugPlayServiceType
Type: REG_DWORD
Data: 0x1

Value 4
Name: RequestedSystemResources
Type: REG_RESOURCE_REQUIREMENTS_LIST
Data:

```

```

Interface Type: Internal
Bus Number: 0
Slot Number: 0
List 0
Descriptor 0
Resource: Interrupt
Option: 0x00000000
Disposition: Shared
Type: Level
Minimum Vector: 0x25
Maximum Vector: 0x25
Descriptor 1
Resource: Port
Option: 0x00000001
Disposition: Device
Type: Port
Length: 0x100
Alignment: 0x100
Minimum Address: 0x00006400
Maximum Address: 0x000064ff
Descriptor 2
Resource: Port
Option: 0x00000008
Disposition: Device
Type: Port
Length: 0x100
Alignment: 0x100
Minimum Address: 0x00006400
Maximum Address: 0x000064ff
Descriptor 3
Resource: Memory
Option: 0x00000001
Disposition: Device
Type: Read / Write
Length: 0x100
Alignment: 0x100
Minimum Address: 0xf76d0000
Maximum Address: 0xf76d00ff
Descriptor 4
Resource: Memory
Option: 0x00000008
Disposition: Device
Type: Read / Write
Length: 0x100
Alignment: 0x100
Minimum Address: 0xf76d0000
Maximum Address: 0xf76d00ff
Descriptor 5
Resource: Memory
Option: 0x00000008
Disposition: Device
Type: Read / Write

```

```

Length: 0x100
Alignment: 0x100
Minimum Address: 0xf76c0000
Maximum Address: 0xf76c0fff
Descriptor 6
Resource: Memory
Option: 0x00000001
Disposition: Device
Type: Read / Write
Length: 0x1000
Alignment: 0x1000
Minimum Address: 0xf76c0000
Maximum Address: 0xf76c0fff
Descriptor 7
Resource: Memory
Option: 0x00000008
Disposition: Device
Type: Read / Write
Length: 0x1000
Alignment: 0x1000
Minimum Address: 0xf76d0000
Maximum Address: 0xf76d00ff
Descriptor 8
Resource: Memory
Option: 0x00000008
Disposition: Device
Type: Read / Write
Length: 0x1000
Alignment: 0x1000
Minimum Address: 0xf76c0000
Maximum Address: 0xf76c0fff
Value 5
Name: Start
Type: REG_DWORD
Data: 0
Value 6
Name: Tag
Type: REG_DWORD
Data: 0x1a
Value 7
Name: Type
Type: REG_DWORD
Data: 0x1
Key Name:
SYSTEM\CurrentControlSet\Services\symc810\Enum
Class Name: <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name: 0
Type: REG_SZ

```

Data: Root\LEGACY_SYMC810\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\SysMgmt
 Class Name: <NO CLASS>
 Last Write Time: 1/6/99 - 1:39 PM

Value 0
 Name: DisplayName
 Type: REG_SZ
 Data: Compaq System Management Driver

Value 1
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: Group
 Type: REG_SZ
 Data: System Management

Value 3
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: SYSTEM32\DRIVERS\SYSMGMT.SYS

Value 4
 Name: InstalledBySetup
 Type: REG_DWORD
 Data: 0x1

Value 5
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 6
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\SysMgmt\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_SYSMGMT\0000

Value 1

Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\SysMgmt\Linkage
 Class Name: <NO CLASS>
 Last Write Time: 1/6/99 - 1:39 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\SysMgmt\Linkage\Dis
 abled
 Class Name: <NO CLASS>
 Last Write Time: 1/6/99 - 1:39 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\SysMgmt\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 1/6/99 - 1:39 PM

Value 0
 Name: DriverMode
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\SysMgmt\Security
 Class Name: <NO CLASS>
 Last Write Time: 1/6/99 - 1:39 PM

Value 0
 Name: Security
 Type: REG_BINARY
 Data:
 00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
 00 00 00
 00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
 80 18 00 4.....
 00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
 00 18 00
 00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
 00 00 00
 00000050 76 00 63 00 00 00 1c 00 - fd 01 02 00 01
 02 00 00 v.c.....
 00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 00
 00 16 00#.....
 00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
 00 00 05
 00000080 20 00 00 00 20 02 00 00 - 00 00 16 00 00
 00 1c 00
 00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
 00 00 00
 000000a0 25 02 00 00 00 00 16 00 - 00 00 18 00 fd
 01 02 00 %.....
 000000b0 01 01 00 00 00 00 05 - 12 00 00 00 25
 02 00 00%...

000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
 01 00 00
 000000d0 00 00 00 05 12 00 00 00 -

Key Name:
 SYSTEM\CurrentControlSet\Services\T128
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x12

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\T128\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\T128\Parameters\Dev
 ice0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DriverParameter
 Type: REG_SZ
 Data: IRQ=5

Key Name:
 SYSTEM\CurrentControlSet\Services\T13B
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group

```

Type:          REG_SZ
Data:          SCSI miniport

Value 2
Name:          Start
Type:          REG_DWORD
Data:          0x4

Value 3
Name:          Tag
Type:          REG_DWORD
Data:          0x13

Value 4
Name:          Type
Type:          REG_DWORD
Data:          0x1

Key Name:
SYSTEM\CurrentControlSet\Services\T13B\Parameters
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\T13B\Parameters\Dev
ice0
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          DriverParameter
Type:          REG_SZ
Data:          IRQ=5

Key Name:
SYSTEM\CurrentControlSet\Services\TapiSrv
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          DisplayName
Type:          REG_SZ
Data:          Telephony Service

Value 1
Name:          ErrorControl
Type:          REG_DWORD
Data:          0x1

Value 2
Name:          ImagePath
Type:          REG_EXPAND_SZ
Data:          %SystemRoot%\system32\tapisrv.exe

Value 3
Name:          ObjectName
Type:          REG_SZ
Data:          LocalSystem

Value 4
Name:          Start
Type:          REG_DWORD
Data:          0x3

```

```

Value 5
Name:          Type
Type:          REG_DWORD
Data:          0x10

Key Name:
SYSTEM\CurrentControlSet\Services\TapiSrv\Performance
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          Close
Type:          REG_SZ
Data:          CloseTapiPerformanceData

Value 1
Name:          Collect
Type:          REG_SZ
Data:          CollectTapiPerformanceData

Value 2
Name:          Library
Type:          REG_SZ
Data:          tapiperf.dll

Value 3
Name:          Open
Type:          REG_SZ
Data:          OpenTapiPerformanceData

Key Name:
SYSTEM\CurrentControlSet\Services\TapiSrv\Security
Class Name:    <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
Name:          Security
Type:          REG_BINARY
Data:          00000000 01 00 14 80 b4 00 00 00 - c0 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 80 00 - 05 00 00 00 00
03 18 00 .....
00000040 8d 00 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 00 00 00 00 03 18 00 - ff 01 0f 00 01
02 00 00 .....
00000060 00 00 00 05 20 00 00 00 - 20 02 00 00 00
03 18 00 .....
00000070 8f 00 02 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
00000080 23 02 00 00 00 03 18 00 - 9d 00 00 00 01
01 00 00 #.....
00000090 00 00 00 05 04 00 00 00 - 23 02 00 00 00
03 18 00 .....#.....
000000a0 9d 00 00 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....

```

```

000000b0 21 02 00 00 01 01 00 00 - 00 00 00 05 12
00 00 00 !.....
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00
.....

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          DisplayName
Type:          REG_SZ
Data:          TCP/IP Service

Value 1
Name:          ErrorControl
Type:          REG_DWORD
Data:          0x1

Value 2
Name:          Group
Type:          REG_SZ
Data:          PNP_TDI

Value 3
Name:          ImagePath
Type:          REG_EXPAND_SZ
Data:          \SystemRoot\System32\drivers\tcpip.sys

Value 4
Name:          Start
Type:          REG_DWORD
Data:          0x2

Value 5
Name:          Type
Type:          REG_DWORD
Data:          0x1

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Enum
Class Name:    <NO CLASS>
Last Write Time: 2/5/99 - 6:32 PM
Value 0
Name:          0
Type:          REG_SZ
Data:          Root\LEGACY_TCPIP\0000

Value 1
Name:          Count
Type:          REG_DWORD
Data:          0x1

Value 2
Name:          NextInstance
Type:          REG_DWORD
Data:          0x1

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Linkage
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: Bind
Type: REG_MULTI_SZ
Data: \Device\N1001

Value 1
Name: Export
Type: REG_MULTI_SZ
Data: \Device\Tcpip\N1001

Value 2
Name: Route
Type: REG_MULTI_SZ
Data: "N100" "N1001"

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Linkage\Disab
led
Class Name: GenericClass
Last Write Time: 2/2/99 - 11:54 AM
Value 0
Name: Bind
Type: REG_MULTI_SZ
Data:

Value 1
Name: Export
Type: REG_MULTI_SZ
Data:

Value 2
Name: Route
Type: REG_MULTI_SZ
Data:

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Parameters
Class Name: GenericClass
Last Write Time: 2/8/99 - 10:11 AM
Value 0
Name: DataBasePath
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\drivers\etc

Value 1
Name: Domain
Type: REG_SZ
Data:

Value 2
Name: EnableSecurityFilters
Type: REG_DWORD
Data: 0

```

```

Value 3
Name: ForwardBroadcasts
Type: REG_DWORD
Data: 0

Value 4
Name: Hostname
Type: REG_SZ
Data: javelina

Value 5
Name: IPEnableRouter
Type: REG_DWORD
Data: 0

Value 6
Name: KeepAliveTime
Type: REG_DWORD
Data: 0x493e0

Value 7
Name: NameServer
Type: REG_SZ
Data:

Value 8
Name: SearchList
Type: REG_SZ
Data:

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Pe
rsistentRoutes
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Wi
nsock
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: HelperDllName
Type: REG_EXPAND_SZ
Data: %SystemRoot%\System32\wshtcpip.dll

Value 1
Name: Mapping
Type: REG_BINARY
Data:
00000000 0b 00 00 00 03 00 00 00 - 02 00 00 00 01
00 00 00 .....
00000010 06 00 00 00 02 00 00 00 - 01 00 00 00 00
00 00 00 .....
00000020 02 00 00 00 00 00 00 00 - 06 00 00 00 00
00 00 00 .....
00000030 00 00 00 00 06 00 00 00 - 00 00 00 00 01
00 00 00 .....
00000040 06 00 00 00 02 00 00 00 - 02 00 00 00 11
00 00 00 .....
00000050 02 00 00 00 02 00 00 00 - 00 00 00 00 02
00 00 00 .....

```

```

00000060 00 00 00 00 11 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000070 11 00 00 00 00 00 00 00 - 02 00 00 00 11
00 00 00 .....
00000080 02 00 00 00 03 00 00 00 - 00 00 00 00
.....

Value 2
Name: MaxSockAddrLength
Type: REG_DWORD
Data: 0x10

Value 3
Name: MinSockAddrLength
Type: REG_DWORD
Data: 0x10

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Performance
Class Name: GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Close
Type: REG_SZ
Data: CloseTcpIpPerformanceData

Value 1
Name: Collect
Type: REG_SZ
Data: CollectTcpIpPerformanceData

Value 2
Name: Library
Type: REG_SZ
Data: Perfctrs.dll

Value 3
Name: Open
Type: REG_SZ
Data: OpenTcpIpPerformanceData

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\Security
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name: Security
Type: REG_BINARY
Data:
00000000 01 00 14 80 c0 00 00 00 - cc 00 00 00 14
00 00 00 .....
00000010 34 00 00 00 02 00 20 00 - 01 00 00 00 02
80 18 00 4.....
00000020 ff 01 0f 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000030 20 02 00 00 02 00 8c 00 - 05 00 00 00 00
00 18 00 .....
00000040 8d 01 02 00 01 01 00 00 - 00 00 00 01 00
00 00 00 .....
00000050 00 01 08 00 00 00 1c 00 - fd 01 02 00 01
02 00 00 .....

```

```

00000060 00 00 00 05 20 00 00 00 - 23 02 00 00 73
45 72 76 ....#...sErv
00000070 00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00
00 00 05 .....
00000080 20 00 00 00 20 02 00 00 - 73 45 72 76 00
00 1c 00 ...sErv...
00000090 ff 01 0f 00 01 02 00 00 - 00 00 00 05 20
00 00 00 .....
000000a0 25 02 00 00 73 45 72 76 - 00 00 18 00 fd
01 02 00 %...sErv.....
000000b0 01 01 00 00 00 00 00 05 - 12 00 00 00 25
02 00 00 .....%...
000000c0 01 01 00 00 00 00 00 05 - 12 00 00 00 01
01 00 00 .....
000000d0 00 00 00 05 12 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\Tcpip\ServiceProvid
er
Class Name:      GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
  Name:          Class
  Type:          REG_DWORD
  Data:          0x8
Value 1
  Name:          DnsPriority
  Type:          REG_DWORD
  Data:          0x7d0
Value 2
  Name:          HostsPriority
  Type:          REG_DWORD
  Data:          0x1f4
Value 3
  Name:          LocalPriority
  Type:          REG_DWORD
  Data:          0x1f3
Value 4
  Name:          Name
  Type:          REG_SZ
  Data:          TCF/IP
Value 5
  Name:          NetbtPriority
  Type:          REG_DWORD
  Data:          0x7d1
Value 6
  Name:          ProviderPath
  Type:          REG_EXPAND_SZ
  Data:          %SystemRoot%\System32\wsock32.dll
Key Name:
SYSTEM\CurrentControlSet\Services\tga
Class Name:      <NO CLASS>
Last Write Time: 12/11/98 - 12:07 PM

```

```

Value 0
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0
Value 1
  Name:          Group
  Type:          REG_SZ
  Data:          Video
Value 2
  Name:          Start
  Type:          REG_DWORD
  Data:          0x4
Value 3
  Name:          Type
  Type:          REG_DWORD
  Data:          0x1
Key Name:
SYSTEM\CurrentControlSet\Services\tga\Device0
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:          InstalledDisplayDrivers
  Type:          REG_MULTI_SZ
  Data:          tga
Value 1
  Name:          VgaCompatible
  Type:          REG_DWORD
  Data:          0
Key Name:
SYSTEM\CurrentControlSet\Services\tml1
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0x1
Value 1
  Name:          Group
  Type:          REG_SZ
  Data:          SCSI miniport
Value 2
  Name:          Start
  Type:          REG_DWORD
  Data:          0x4
Value 3
  Name:          Tag
  Type:          REG_DWORD
  Data:          0x18
Value 4
  Name:          Type

```

```

Type:          REG_DWORD
Data:          0x1
Key Name:
SYSTEM\CurrentControlSet\Services\tml1\Parameters
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Key Name:
SYSTEM\CurrentControlSet\Services\tml1\Parameters\Dev
ice0
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:          DriverParameter
  Type:          REG_SZ
  Data:          IRQ=15
Key Name:
SYSTEM\CurrentControlSet\Services\Ultra124
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0x1
Value 1
  Name:          Group
  Type:          REG_SZ
  Data:          SCSI miniport
Value 2
  Name:          Start
  Type:          REG_DWORD
  Data:          0x4
Value 3
  Name:          Tag
  Type:          REG_DWORD
  Data:          0xb
Value 4
  Name:          Type
  Type:          REG_DWORD
  Data:          0x1
Key Name:
SYSTEM\CurrentControlSet\Services\Ultra14f
Class Name:      <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM
Value 0
  Name:          ErrorControl
  Type:          REG_DWORD
  Data:          0x1
Value 1
  Name:          Group
  Type:          REG_SZ
  Data:          SCSI miniport

```

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0xa

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Ultra24f
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0xc

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\update
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 3:41 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Base

Value 2
 Name: Start

Type: REG_DWORD
 Data: 0x1

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\update\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: Count
 Type: REG_DWORD
 Data: 0

Value 1
 Name: NextInstance
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\UPS
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: %SystemRoot%\System32\ups.exe

Value 2
 Name: ObjectName
 Type: REG_SZ
 Data: LocalSystem

Value 3
 Name: Start
 Type: REG_DWORD
 Data: 0x3

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x10

Key Name:
 SYSTEM\CurrentControlSet\Services\v7vram
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\v7vram\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: vga
 vga256

Value 1
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\VgaSave
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video Save

Value 2
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: \SystemRoot\System32\drivers\vga.sys

Value 3
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 5
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\VgaSave\Device0
 Class Name: <NO CLASS>
 Last Write Time: 1/6/99 - 1:36 PM

Value 0
 Name: HardwareInformation.Crc32
 Type: REG_BINARY
 Data: 00000000 40 30 51 36
 @0Q6

Value 1
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: vga

Value 2
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\VgaSave\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:32 PM

Value 0
 Name: 0
 Type: REG_SZ
 Data: Root\LEGACY_VGASAVE\0000

Value 1
 Name: Count
 Type: REG_DWORD
 Data: 0x1

Value 2
 Name: NextInstance
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\VgaStart
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video Init

Value 2
 Name: ImagePath
 Type: REG_EXPAND_SZ
 Data: \SystemRoot\System32\drivers\vga.sys

Value 3
 Name: Start
 Type: REG_DWORD
 Data: 0x1

Value 4
 Name: Tag
 Type: REG_DWORD
 Data: 0x1

Value 5
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\VgaStart\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ForceVga
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: vga

Value 2
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\VgaStart\Enum
 Class Name: <NO CLASS>
 Last Write Time: 2/5/99 - 6:33 PM

Value 0
 Name: Count
 Type: REG_DWORD
 Data: 0

Value 1
 Name: NextInstance
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\VXD
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:17 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\VXD\JAVASUP
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:17 PM

Value 0
 Name: Start
 Type: REG_BINARY
 Data: 00000000 00
 .

Value 1
 Name: StaticVxD
 Type: REG_SZ
 Data: JAVASUP.VXD

Key Name:
 SYSTEM\CurrentControlSet\Services\w3svc
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\w3svc\parameters
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\w3svc\parameters\AD
 CLaunch
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\w3svc\parameters\AD
 CLaunch\AdvancedDataFactory
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\w3svc\parameters\AD
 CLaunch\RDSServer.DataFactory
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Key Name:
 SYSTEM\CurrentControlSet\Services\w3svc\parameters\Vi
 rtual Roots
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:31 PM

Value 0
 Name: /MSADC
 Type: REG_SZ
 Data: C:\Program Files\Common
 Files\System\MSADC,,5

Key Name:
 SYSTEM\CurrentControlSet\Services\Wd33c93
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Group
 Type: REG_SZ
 Data: SCSI miniport

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Tag
 Type: REG_DWORD
 Data: 0x11

Value 4
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Wd33c93\Parameters
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\Wd33c93\Parameters\
 Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: DisableDisconnect
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: DriverParameter
 Type: REG_SZ
 Data: IRQ=10;DMA=6

Key Name:
 SYSTEM\CurrentControlSet\Services\wd90c24a
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: REG_SZ

Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\wd90c24a\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: vga
 vga256
 wd90c24a

Value 1
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\wdvga
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\wdvga\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ

Data: vga
 vga256
 vga64k

Value 1
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\weitekp9
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\weitekp9\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM

Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: weitekp9

Value 1
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0

Key Name:
 SYSTEM\CurrentControlSet\Services\WinSock
 Class Name: GenericClass
 Last Write Time: 12/11/98 - 6:04 AM

Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0x1

Value 1
 Name: Start


```

Type:          REG_DWORD
Data:          0x3

Value 2
Name:          Type
Type:          REG_DWORD
Data:          0x4

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Autodial
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 4:16 PM
Value 0
Name:          AutodialDllName32
Type:          REG_SZ
Data:          wininet.dll

Value 1
Name:          AutodialFonName32
Type:          REG_SZ
Data:          InternetAutodialCallback

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Linkage
Class Name:    GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Linkage\Dis
abled
Class Name:    GenericClass
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Parameters
Class Name:    GenericClass
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          Transports
Type:          REG_MULTI_SZ
Data:          Tcpip
NetBIOS

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Setup
Migration
Class Name:    <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM
Value 0
Name:          Known Static Providers
Type:          REG_MULTI_SZ
Data:          Tcpip
NwlnkIpx
NwlnkSpx
AppleTalk
IsoTp

Value 1

```

```

Name:          Provider List
Type:          REG_MULTI_SZ
Data:          Tcpip
NetBIOS

Value 2
Name:          Setup Version
Type:          REG_DWORD
Data:          0x1009

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Setup
Migration\Providers
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Setup
Migration\Providers\NetBIOS
Class Name:    <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM
Value 0
Name:          WinSock 1.1 Provider Data
Type:          REG_BINARY
Data:          00000000 0e 10 00 00 11 00 00 00 - 14 00 00 00 14
00 00 00 .....
00000010 05 00 00 00 00 00 00 80 - 00 fa 00 00 68
00 00 00 .....h...
00000020 09 12 00 00 11 00 00 00 - 14 00 00 00 14
00 00 00 .....
00000030 02 00 00 00 00 00 00 80 - 00 fa 00 00 40
00 00 00 .....@...
00000040 5c 00 44 00 65 00 76 00 - 69 00 63 00 65
00 5c 00 \.D.e.v.i.c.e.\.
00000050 4e 00 65 00 74 00 42 00 - 54 00 5f 00 4e
00 31 00 N.e.t.B.T._.N.l.
00000060 30 00 30 00 31 00 00 00 - 5c 00 44 00 65
00 76 00 0.0.1...\.D.e.v.
00000070 69 00 63 00 65 00 5c 00 - 4e 00 65 00 74
00 42 00 i.c.e.\.N.e.t.B.
00000080 54 00 5f 00 4e 00 31 00 - 30 00 30 00 31
00 00 00 T._.N.l.0.0.1...

Value 1
Name:          WinSock 2.0 Provider ID
Type:          REG_BINARY
Data:          00000000 30 18 5f 8d 73 c2 cf 11 - 95 c8 00 80 5f
48 a1 92 0._.s.....H..

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Setup
Migration\Providers\Tcpip
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          WinSock 2.0 Provider ID
Type:          REG_BINARY
Data:

```

```

00000000 a0 1a 0f e7 8b ab cf 11 - 8c a3 00 80 5f
48 a1 92 ....._H..

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock\Setup
Migration\Well Known Guides
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          AppleTalk
Type:          REG_BINARY
Data:          00000000 a0 17 3b 2c df c6 cf 11 - 95 c8 00 80 5f
48 a1 92 ..;.....H..

Value 1
Name:          IsoTp
Type:          REG_BINARY
Data:          00000000 b0 cb e4 89 c1 b9 cf 11 - 95 c8 00 80 5f
48 a1 92 ....._H..

Value 2
Name:          McsXns
Type:          REG_BINARY
Data:          00000000 b1 cb e4 89 c1 b9 cf 11 - 95 c8 00 80 5f
48 a1 92 ....._H..

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:02 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          Current_NameSpace_Catalog
Type:          REG_SZ
Data:          NameSpace_Catalog5

Value 1
Name:          Current_Protocol_Catalog
Type:          REG_SZ
Data:          Protocol_Catalog9

Value 2
Name:          WinSock_Registry_Version
Type:          REG_SZ
Data:          2.0

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\NameSpace_Catalog5
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          Next_Provider_ID

```

```

Type:          REG_DWORD
Data:          0x7d0

Value 1
Name:          Num_Catalog_Entries
Type:          REG_DWORD
Data:          0x1

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\NameSpace_Catalog5\Catalog_Entries
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\NameSpace_Catalog5\Catalog_Entries\000000000001
Class Name:    <NO CLASS>
Last Write Time: 12/11/98 - 6:04 AM
Value 0
Name:          DisplayString
Type:          REG_SZ
Data:          TCP/IP

Value 1
Name:          Enabled
Type:          REG_DWORD
Data:          0x1

Value 2
Name:          LibraryPath
Type:          REG_SZ
Data:          %SystemRoot%\System32\rnr20.dll

Value 3
Name:          ProviderId
Type:          REG_BINARY
Data:          40 9d 05 22 9e 7e cf 11 - ae 5a 00 aa 00
a7 11 2b @..".~...Z.....+

Value 4
Name:          StoresServiceClassInfo
Type:          REG_DWORD
Data:          0x520000

Value 5
Name:          SupportedNameSpace
Type:          REG_DWORD
Data:          0xc

Value 6
Name:          Version
Type:          REG_DWORD
Data:          0

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\Protocol_Catalog9
Class Name:    <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM

```

```

Value 0
Name:          Next_Catalog_Entry_ID
Type:          REG_DWORD
Data:          0x408

Value 1
Name:          Next_Provider_ID
Type:          REG_DWORD
Data:          0x1

Value 2
Name:          Num_Catalog_Entries
Type:          REG_DWORD
Data:          0x5

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\Protocol_Catalog9\Catalog_Entries
Class Name:    <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\Protocol_Catalog9\Catalog_Entries\000000000001
Class Name:    <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM

Value 0
Name:          PackedCatalogItem
Type:          REG_BINARY
Data:          00000000 25 53 79 73 74 65 6d 52 - 6f 6f 74 25 5c
73 79 73 %SystemRoot%\sys
00000010 74 65 6d 33 32 5c 6d 73 - 61 66 64 2e 64
6c 6c 00 tem32\msafd.dll.
00000020 28 63 75 74 29 0d 0a 29 - 0d 0a 28 20 28
74 6f 73 (cut)..( (tos
00000030 74 72 69 6e 67 20 58 20 - 59 29 0d 0a 20
20 20 20 tring X Y)..
00000040 28 61 74 6f 6d 20 58 29 - 0d 0a 20 20 20
20 28 63 (atom X).. (c
00000050 75 74 29 0d 0a 20 20 20 - 20 28 73 74 72
69 6e 67 ut).. (string
00000060 5f 66 72 6f 6d 20 58 20 - 59 29 0d 0a 29
0d 0a 28 _from X Y)..(
00000070 20 28 74 6f 73 74 72 69 - 6e 67 20 58 20
59 29 0d (tostring X Y)..
00000080 0a 20 20 20 20 28 69 6e - 74 65 67 65 72
20 58 29 . (integer X)
00000090 0d 0a 20 20 20 20 28 63 - 75 74 29 0d 0a
20 20 20 .. (cut)..
000000a0 20 28 73 74 72 69 6e 67 - 5f 66 72 6f 6d
20 58 20 (string_from X
000000b0 59 29 0d 0a 29 0d 0a 28 - 20 28 74 6f 73
74 72 69 Y)..( (tostr
000000c0 6e 67 20 58 20 22 3c 5f - 76 61 72 5f 3e
22 29 0d ng X "<_var_>").
000000d0 0a 20 20 20 20 28 76 61 - 72 20 58 29 0d
0a 29 0d . (var X)..)
000000e0 0a 28 70 72 69 6e 74 61 - 6c 6c 20 28 29
29 0d 0a .(printall ())..
000000f0 0d 0a 28 20 28 70 72 69 - 6e 74 61 6c 6c
20 28 48 ..( (printall (H

```

```

00000100 7c 54 29 29 66 00 02 00 - 00 00 00 00 00
00 00 00 (T))f.....
00000110 00 00 00 00 08 00 00 00 - a0 1a 0f e7 8b
ab cf 11 .....
00000120 8c a3 00 80 5f 48 a1 92 - e9 03 00 00 01
00 00 00 ....H.....
00000130 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000140 00 00 00 00 00 00 00 00 - 00 00 00 00 02
00 00 00 .....
00000150 02 00 00 00 10 00 00 00 - 10 00 00 00 01
00 00 00 .....
00000160 06 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000170 00 00 00 00 00 00 00 00 - 4d 00 53 00 41
00 46 00 .....M.S.A.F.
00000180 44 00 20 00 54 00 63 00 - 70 00 69 00 70
00 20 00 D..T.c.p.i.p..
00000190 5b 00 54 00 43 00 50 00 - 2f 00 49 00 50
00 5d 00 [.T.C.P./I.P.].
000001a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000200 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000210 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000220 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000230 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000240 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000250 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000260 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000270 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000280 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000290 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....

```

```

000002f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000300 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000310 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000320 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000330 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000340 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000350 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000360 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000370 00 00 00 00 00 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\Protocol_Catalog9\Catalog_Entries\000000000002
Class Name: <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM
Value 0
Name: PackedCatalogItem
Type: REG_BINARY
Data:
00000000 25 53 79 73 74 65 6d 52 - 6f 6f 74 25 5c
73 79 73 %SystemRoot%\sys
00000010 74 65 6d 33 32 5c 6d 73 - 61 66 64 2e 64
6c 6c 00 tem32\msafd.dll.
00000020 28 63 75 74 29 0d 0a 29 - 0d 0a 28 20 28
74 6f 73 (cut)...( (tos
00000030 74 72 69 6e 67 20 58 20 - 59 29 0d 0a 20
20 20 20 tring X Y)..
00000040 28 61 74 6f 6d 20 58 29 - 0d 0a 20 20 20
20 28 63 (atom X).. (c
00000050 75 74 29 0d 0a 20 20 20 - 20 28 73 74 72
69 6e 67 ut)... (string
00000060 5f 66 72 6f 6d 20 58 20 - 59 29 0d 0a 29
0d 0a 28 from X Y)...(
00000070 20 28 74 6f 73 74 72 69 - 6e 67 20 58 20
59 29 0d (tostring X Y)..
00000080 0a 20 20 20 20 28 69 6e - 74 65 67 65 72
20 58 29 . (integer X)
00000090 0d 0a 20 20 20 20 28 63 - 75 74 29 0d 0a
20 20 20 . (cut)..
000000a0 20 28 73 74 72 69 6e 67 - 5f 66 72 6f 6d
20 58 20 (string_from X
000000b0 59 29 0d 0a 29 0d 0a 28 - 20 28 74 6f 73
74 72 69 Y)...( (tostrin
000000c0 6e 67 20 58 20 22 3c 5f - 76 61 72 5f 3e
22 29 0d ng X "<var>").
000000d0 0a 20 20 20 20 28 76 61 - 72 20 58 29 0d
0a 29 0d . (var X)..
000000e0 0a 28 70 72 69 6e 74 61 - 6c 6c 20 28 29
29 0d 0a .(printall ()).
000000f0 0d 0a 28 20 28 70 72 69 - 6e 74 61 6c 6c
20 28 48 ..( (printall (H
00000100 7c 54 29 29 09 06 02 00 - 00 00 00 00 00
00 00 00 |T)).....

```

```

00000110 00 00 00 00 08 00 00 00 - a0 1a 0f e7 8b
ab cf 11 .....
00000120 8c a3 00 80 5f 48 a1 92 - ea 03 00 00 01
00 00 00 ....._H.....
00000130 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000140 00 00 00 00 00 00 00 00 - 00 00 00 00 02
00 00 00 .....
00000150 02 00 00 00 10 00 00 00 - 10 00 00 00 02
00 00 00 .....
00000160 11 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000170 bb ff 00 00 00 00 00 00 - 4d 00 53 00 41
00 46 00 .....M.S.A.F.
00000180 44 00 20 00 54 00 63 00 - 70 00 69 00 70
00 20 00 D..T.c.p.i.p..
00000190 5b 00 55 00 44 00 50 00 - 2f 00 49 00 50
00 5d 00 [..U.D.P./..I.P.].
000001a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000001f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000200 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000210 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000220 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000230 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000240 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000250 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000260 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000270 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000280 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000290 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
000002f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....

```

```

00000300 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000310 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000320 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000330 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000340 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000350 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000360 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .....
00000370 00 00 00 00 00 00 00 00 -
.....

```

```

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\Protocol_Catalog9\Catalog_Entries\000000000003
Class Name: <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM
Value 0
Name: PackedCatalogItem
Type: REG_BINARY
Data:
00000000 25 53 79 73 74 65 6d 52 - 6f 6f 74 25 5c
73 79 73 %SystemRoot%\sys
00000010 74 65 6d 33 32 5c 6d 73 - 61 66 64 2e 64
6c 6c 00 tem32\msafd.dll.
00000020 28 63 75 74 29 0d 0a 29 - 0d 0a 28 20 28
74 6f 73 (cut)...( (tos
00000030 74 72 69 6e 67 20 58 20 - 59 29 0d 0a 20
20 20 20 tring X Y)..
00000040 28 61 74 6f 6d 20 58 29 - 0d 0a 20 20 20
20 28 63 (atom X).. (c
00000050 75 74 29 0d 0a 20 20 20 - 20 28 73 74 72
69 6e 67 ut).. (string
00000060 5f 66 72 6f 6d 20 58 20 - 59 29 0d 0a 29
0d 0a 28 from X Y)...(
00000070 20 28 74 6f 73 74 72 69 - 6e 67 20 58 20
59 29 0d (tostring X Y)..
00000080 0a 20 20 20 20 28 69 6e - 74 65 67 65 72
20 58 29 . (integer X)
00000090 0d 0a 20 20 20 20 28 63 - 75 74 29 0d 0a
20 20 20 .. (cut)..
000000a0 20 28 73 74 72 69 6e 67 - 5f 66 72 6f 6d
20 58 20 (string_from X
000000b0 59 29 0d 0a 29 0d 0a 28 - 20 28 74 6f 73
74 72 69 Y)...( (tostrin
000000c0 6e 67 20 58 20 22 3c 5f - 76 61 72 5f 3e
22 29 0d ng X "<var>").
000000d0 0a 20 20 20 20 28 76 61 - 72 20 58 29 0d
0a 29 0d . (var X)..
000000e0 0a 28 70 72 69 6e 74 61 - 6c 6c 20 28 29
29 0d 0a .(printall ()).
000000f0 0d 0a 28 20 28 70 72 69 - 6e 74 61 6c 6c
20 28 48 ..( (printall (H
00000100 7c 54 29 29 09 06 02 00 - 00 00 00 00 00
00 00 00 |T)).....
00000110 00 00 00 00 0c 00 00 00 - a0 1a 0f e7 8b
ab cf 11 .....

```

00000120 8c a3 00 80 5f 48 a1 92 - eb 03 00 00 01
00 00 00H.....
00000130 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000140 00 00 00 00 00 00 00 00 - 00 00 00 00 02
00 00 00
00000150 02 00 00 00 10 00 00 00 - 10 00 00 00 03
00 00 00
00000160 00 00 00 00 ff 00 00 00 - 00 00 00 00 00
00 00 00
00000170 bb ff 00 00 00 00 00 00 - 4d 00 53 00 41
00 46 00M.S.A.F.
00000180 44 00 20 00 54 00 63 00 - 70 00 69 00 70
00 20 00 D..T.c.p.i.p..
00000190 5b 00 52 00 41 00 57 00 - 2f 00 49 00 50
00 5d 00 [.R.A.W./I.P.]
000001a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000001b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000001c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000001d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000001e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000001f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000200 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000210 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000220 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000230 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000240 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000250 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000260 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000270 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000280 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000290 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000300 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00

00000310 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000320 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000330 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000340 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000350 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000360 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000370 00 00 00 00 00 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\Protocol_Catalog9\Catalog_Entries\000000000004
Class Name: <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM
Value 0
Name: PackedCatalogItem
Type: REG_BINARY
Data:
00000000 25 53 79 73 74 65 6d 52 - 6f 6f 74 25 5c
73 79 73 %SystemRoot%\sys
00000010 74 65 6d 33 32 5c 6d 73 - 61 66 64 2e 64
6c 6c 00 tem32\msafd.dll.
00000020 28 63 75 74 29 0d 0a 29 - 0d 0a 28 20 28
74 6f 73 (cut)..((tos
00000030 74 72 69 6e 67 20 58 20 - 59 29 0d 0a 20
20 20 20 tring X Y)..
00000040 28 61 74 6f 6d 20 58 29 - 0d 0a 20 20 20
20 28 63 (atom X).. (c
00000050 75 74 29 0d 0a 20 20 20 - 20 28 73 74 72
69 6e 67 ut).. (string
00000060 5f 66 72 6f 6d 20 58 20 - 59 29 0d 0a 29
0d 0a 28 _from X Y)..((to
00000070 20 28 74 6f 73 74 72 69 - 6e 67 20 58 20
59 29 0d (tostring X Y).
00000080 0a 20 20 20 20 28 69 6e - 74 65 67 65 72
20 58 29 . (integer X)
00000090 0d 0a 20 20 20 20 28 63 - 75 74 29 0d 0a
20 20 20 .. (cut)..
000000a0 20 28 73 74 72 69 6e 67 - 5f 66 72 6f 6d
20 58 20 (string_from X
000000b0 59 29 0d 0a 29 0d 0a 28 - 20 28 74 6f 73
74 72 69 Y)..((tostr
000000c0 6e 67 20 58 20 22 3c 5f - 76 61 72 5f 3e
22 29 0d ng X "< var >").
000000d0 0a 20 20 20 20 28 76 61 - 72 20 58 29 0d
0a 29 0d . (var X)..
000000e0 0a 28 70 72 69 6e 74 61 - 6c 6c 20 28 29
29 0d 0a .(printall ())..
000000f0 0d 0a 28 20 28 70 72 69 - 6e 74 61 6c 6c
20 28 48 ..((printall (H
00000100 7c 54 29 29 0e 00 02 00 - 00 00 00 00 00
00 00 00 |T)).....
00000110 00 00 00 00 08 00 00 00 - 30 18 5f 8d 73
c2 cf 110..s...
00000120 95 c8 00 80 5f 48 a1 92 - 06 04 00 00 01
00 00 00H.....

00000130 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000140 00 00 00 00 00 00 00 00 - 00 00 00 00 02
00 00 00
00000150 11 00 00 00 14 00 00 00 - 14 00 00 00 05
00 00 00
00000160 00 00 00 80 00 00 00 00 - 00 00 00 00 00
00 00 00
00000170 00 fa 00 00 00 00 00 00 - 4d 00 53 00 41
00 46 00M.S.A.F.
00000180 44 00 20 00 4e 00 65 00 - 74 00 42 00 49
00 4f 00 D..N.e.t.B.I.O.
00000190 53 00 20 00 5b 00 5c 00 - 44 00 65 00 76
00 69 00 S..[\.D.e.v.i.
000001a0 63 00 65 00 5c 00 4e 00 - 65 00 74 00 42
00 54 00 c.e.\.N.e.t.B.T.
000001b0 5f 00 4e 00 31 00 30 00 - 30 00 31 00 5d
00 20 00 .N.l.0.0.l.]..
000001c0 53 00 45 00 51 00 50 00 - 41 00 43 00 4b
00 45 00 S.E.Q.P.A.C.K.E.
000001d0 54 00 20 00 30 00 00 00 - 00 00 00 00 00
00 00 00 T..0.....
000001e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000001f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000200 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000210 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000220 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000230 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000240 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000250 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000260 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000270 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000280 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000290 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000300 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000310 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00

00000320 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000330 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000340 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000350 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000360 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000370 00 00 00 00 00 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\WinSock2\Parameters
\Protocol_Catalog9\Catalog_Entries\000000000005
Class Name: <NO CLASS>
Last Write Time: 2/2/99 - 11:55 AM
Value 0
Name: PackedCatalogItem
Type: REG_BINARY
Data:
00000000 25 53 79 73 74 65 6d 52 - 6f 6f 74 25 5c
73 79 73 %SystemRoot%\sys
00000010 74 65 6d 33 32 5c 6d 73 - 61 66 64 2e 64
6c 6c 00 tem32\msafd.dll.
00000020 28 63 75 74 29 0d 0a 29 - 0d 0a 28 20 28
74 6f 73 (cut)...((tos
00000030 74 72 69 6e 67 20 58 20 - 59 29 0d 0a 20
20 20 20 tring X Y)..
00000040 28 61 74 6f 6d 20 58 29 - 0d 0a 20 20 20
20 28 63 (atom X).. (c
00000050 75 74 29 0d 0a 20 20 20 - 20 28 73 74 72
69 6e 67 ut).. (string
00000060 5f 66 72 6f 6d 20 58 20 - 59 29 0d 0a 29
0d 0a 28 from X Y)..((c
00000070 20 28 74 6f 73 74 72 69 - 6e 67 20 58 20
59 29 0d (tostring X Y)..
00000080 0a 20 20 20 20 28 69 6e - 74 65 67 65 72
20 58 29 (integer X)
00000090 0d 0a 20 20 20 28 63 - 75 74 29 0d 0a
20 20 20 (cut)..
000000a0 20 28 73 74 72 69 6e 67 - 5f 66 72 6f 6d
20 58 20 (string_from X
000000b0 59 29 0d 0a 29 0d 0a 28 - 20 28 74 6f 73
74 72 69 Y)..((tostr
000000c0 6e 67 20 58 20 22 3c 5f - 76 61 72 5f 3e
22 29 0d ng X "< var >").
000000d0 0a 20 20 20 20 28 76 61 - 72 20 58 29 0d
0a 29 0d (var X)..
000000e0 0a 28 70 72 69 6e 74 61 - 6c 6c 20 28 29
29 0d 0a (printall ()).
000000f0 0d 0a 28 20 28 70 72 69 - 6e 74 61 6c 6c
20 28 48 ..((printall (H
00000100 7c 54 29 29 09 02 02 00 - 00 00 00 00 00
00 00 00 |T)).....
00000110 00 00 00 00 08 00 00 00 - 30 18 5f 8d 73
c2 cf 110..s..
00000120 95 c8 00 80 5f 48 a1 92 - 07 04 00 00 01
00 00 00H.....
00000130 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00

00000140 00 00 00 00 00 00 00 00 - 00 00 00 00 02
00 00 00
00000150 11 00 00 00 14 00 00 00 - 14 00 00 00 02
00 00 00
00000160 00 00 00 80 00 00 00 00 - 00 00 00 00 00
00 00 00
00000170 00 fa 00 00 00 00 00 00 - 4d 00 53 00 41
00 46 00M.S.A.F.
00000180 44 00 20 00 4e 00 65 00 - 74 00 42 00 49
00 4f 00 D..N.e.t.B.I.O.
00000190 53 00 20 00 5b 00 5c 00 - 44 00 65 00 76
00 69 00 S..[\..D.e.v.i.
000001a0 63 00 65 00 5c 00 4e 00 - 65 00 74 00 42
00 54 00 c.e.\.N.e.t.B.T.
000001b0 5f 00 4e 00 31 00 30 00 - 30 00 31 00 5d
00 20 00 ..N.l.0.0.l.]..
000001c0 44 00 41 00 54 00 41 00 - 47 00 52 00 41
00 4d 00 D.A.T.A.G.R.A.M.
000001d0 20 00 30 00 00 00 00 00 - 00 00 00 00 00
00 00 00 .0.....
000001e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000001f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000200 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000210 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000220 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000230 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000240 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000250 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000260 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000270 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000280 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000290 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002a0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002b0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002c0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002d0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002e0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
000002f0 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000300 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000310 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000320 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00

00000330 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000340 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000350 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000360 00 00 00 00 00 00 00 00 - 00 00 00 00 00
00 00 00
00000370 00 00 00 00 00 00 00 00 -
.....

Key Name:
SYSTEM\CurrentControlSet\Services\WinTrust
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinTrust\SubjectPac
kages
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
SYSTEM\CurrentControlSet\Services\WinTrust\SubjectPac
kages\MS Subjects 1
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:16 PM
Value 0
Name: \$DLL
Type: REG_SZ
Data: C:\WINNT40\System32\MsSip1.dll

Key Name:
SYSTEM\CurrentControlSet\Services\WinTrust\SubjectPac
kages\MS Subjects 2
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:16 PM
Value 0
Name: \$DLL
Type: REG_SZ
Data: C:\WINNT40\System32\MsSip2.dll

Key Name:
SYSTEM\CurrentControlSet\Services\WinTrust\SubjectPac
kages\MS Subjects 3
Class Name: <NO CLASS>
Last Write Time: 12/11/98 - 4:16 PM
Value 0
Name: \$DLL
Type: REG_SZ
Data: C:\WINNT40\System32\MsSip3.dll

Key Name:
SYSTEM\CurrentControlSet\Services\WinTrust\TrustProvi
ders
Class Name: <NO CLASS>
Last Write Time: 10/10/96 - 2:09 AM

Key Name:
 SYSTEM\CurrentControlSet\Services\WinTrust\TrustProv
 iders\Software Publisher
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 4:16 PM
 Value 0
 Name: \$DLL
 Type: REG_SZ
 Data: C:\WINNT40\System32\SoftPub.dll

Key Name:
 SYSTEM\CurrentControlSet\Services\Xga
 Class Name: <NO CLASS>
 Last Write Time: 12/11/98 - 12:07 PM
 Value 0
 Name: ErrorControl
 Type: REG_DWORD
 Data: 0

Value 1
 Name: Group
 Type: REG_SZ
 Data: Video

Value 2
 Name: Start
 Type: REG_DWORD
 Data: 0x4

Value 3
 Name: Type
 Type: REG_DWORD
 Data: 0x1

Key Name:
 SYSTEM\CurrentControlSet\Services\Xga\Device0
 Class Name: <NO CLASS>
 Last Write Time: 10/10/96 - 2:09 AM
 Value 0
 Name: InstalledDisplayDrivers
 Type: REG_MULTI_SZ
 Data: xga
 vga256

Value 1
 Name: Monitor
 Type: REG_SZ
 Data: drivers\vga.bvp

Value 2
 Name: VgaCompatible
 Type: REG_DWORD
 Data: 0

SUT System Configuration

Date 02/03/1999
 Time 17:17:10
 Product PROLIANT 7000
 Machine ID
 From System Board CPQ1601
 Processor Pentium II Xeon(R)
 at 450 MHz
 Slot 1
 Secondary Cache 2048K
 CPU ID 0653
 Processor Pentium II Xeon(R)
 at 450 MHz
 Slot 2
 Secondary Cache 2048K
 CPU ID 0653
 Processor Pentium II Xeon(R)
 at 450 MHz
 Slot 3
 Secondary Cache 2048K
 CPU ID 0653
 Processor Pentium II Xeon(R)
 at 450 MHz
 Slot 4
 Secondary Cache 2048K
 CPU ID 0653
 Processor(s) Mapped Out None
 Numeric Coprocessor Integrated 387-
 Compatible
 Expansion Bus ISA, PCI
 System Identification Number None
 CPU Mode Real Mode
 System ROM
 Revision 10/30/1998
 Family P43
 Flashable Yes
 Supports F10 partition Yes
 Video Controller ROM
 Revision 3.93
 Keyboard Controller ROM
 Revision C.7 06/14/1991
 Family K

Option ROMs
 Address Range C0000 - C7FFF
 Data Dump (1998/06/30 16:36)
 Address Range C8000 - CBFFF
 Data Dump (04/22/98
 SMART-2 Option ROM/BIOS (C)Copyright COMPAQ...)
 Address Range E8000 - EDFFF
 Data Dump (Copyright 1998
 Compaq Computer Corporation-All Rights Reser...)
 Bootblock ROM 04/07/1998
 Standby Recovery Server
 Status Disabled
 COM Port COM1
 Server Configuration Recovery
 Timeout Value 1 minutes
 Memory Boards Identified:
 Expansion Board
 Bank 1 (EDO) 512 Megabytes
 Bank 2 (EDO) 512 Megabytes
 Bank 3 (EDO) 512 Megabytes
 Bank 4 (EDO) 512 Megabytes
 Bank 5 (EDO) 2048 Megabytes
 Bank 6 0 Megabytes
 Bank 7 0 Megabytes
 Bank 8 0 Megabytes
 Total Compaq Memory 4096 Megabytes
 Keyboard Enhanced
 LPT Ports LPT1 (Address 3BC)
 COM Ports COM1 (Address 3F8)
 COM2 (Address 2F8)
 Diskette Drive A 1.44 Megabyte (3.5
 inch)
 Drive Controller 1, Compaq Smart Array 3200 Rev. A
 Controller
 IDA Firmware Revision 3.06
 Array Accelerator Memory 57344 Kbytes
 Reserved for writes 57344 Kbytes
 Accelerator Status Enabled
 Battery count 3
 Batteries charged 3
 Batteries failed 0
 ProLiant Bus 1
 (internal/external), Rev. JB1F
 ProLiant Bus 2 (external),
 Rev. JB1F
 Logical Drive 1 17860 Megabyte
 Fault Tolerance None
 OS Format Multi-Sector
 Distribution
 Drive geometry (Cyl, Hds, Sec) 4275, 255, 32

```

Array Accelerator . . . . . Disabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04126600001845JD2V
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3308219872
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 870817956
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1015869
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . 128
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04135400001845J3JP
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3254707799
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 841117437
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958036
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 845
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04077500001844HM3W
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3993626463
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1396635023
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 961988
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 56
Recovers read failed . . . . 0
Bus faults . . . . . 1

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04145700001845J574
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3800127133
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1105286106
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935599
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 1085
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04112400001845JEAY
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3837638593
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1216817509
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 941278
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 95
Recovers read failed . . . . 0
Bus faults . . . . . 2

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04022900001835H4TE
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299425
Sectors read . . . . . *3716048446
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1088667902
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 929130
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . 727
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ041372000018450CUU
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299425
Sectors read . . . . . *3682723613
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1098214375
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 933352
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1117
Recovers read failed . . . . 0

```

```

Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04137300001845J1BW
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3469637409
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1188667407
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 930917
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 586
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04132100001842K5Q3
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3536793068
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1266139183
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935805
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . 828
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04126300001845J3GX
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3188813343
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1222668486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 939427
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 22
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0389410000184509JH
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3233017791
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1220873045
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 930864
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . 385
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04161100001841HRNK
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3350940077
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1172929466
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 937830
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 696
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04166600001845J555
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299369
Sectors read . . . . . *3568625156
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 985370872
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 937915
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 922
Recovers read failed . . . . 0
Bus faults . . . . . 3

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04153300001845J517
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299397
Sectors read . . . . . *3738373803
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1002300853
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935232
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 929
Recovers read failed . . . . 0

```



```

Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04146800018450BYU
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341144
Sectors read . . . . . *1395569624
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1746883131
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1103498
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 439
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0414020001845HEJQ
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299396
Sectors read . . . . . *25811233
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1252352993
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 932734
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . 1235
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0415240001845HSU5
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 299396
Sectors read . . . . . *4123598281
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1273831207
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 932656
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 174
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1005
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04168200018450BSG
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341144
Sectors read . . . . . *1036345264
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1348088619
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1108865
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 481
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ041673000184507L6
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341144
Sectors read . . . . . *1003555145
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1419563300
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 1102712
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 726
Recovers read failed . . . . 0
Bus faults . . . . . 1

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0414330001838HDK9
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341144
Sectors read . . . . . *1033539280
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1491978914
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1100617
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 983
Recovers read failed . . . . 0
Bus faults . . . . . 2

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04140600018450CVK
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341144
Sectors read . . . . . *886868524
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1418275658
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1097604
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1090
Recovers read failed . . . . 0

```

```

Bus faults . . . . . 4

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ041411000018450BYP
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299418
Sectors read . . . . . *3739073806
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1252906639
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 926507
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1657
Recovers read failed . . . . 0
Bus faults . . . . . 3

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04165300001845JFJV
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299418
Sectors read . . . . . *3683549312
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1228174945
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 925243
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 916
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ0414920000184506ZW
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 299418
Sectors read . . . . . *3503891022
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1230844920
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 925065
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 2114
Recovers read failed . . . . 0
Bus faults . . . . . 0

Logical Drive 2 . . . . . 13627 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 3262, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04126600001845JD2V
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3308219872
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 870817956
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1016045
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 128
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04135400001845J3JP
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378

```

```

Sectors read . . . . . *3254707799
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 841117437
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958212
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 845
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04077500001844HM3W
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3993626463
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1396635023
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 962164
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 56
Recovers read failed . . . . 0
Bus faults . . . . . 1

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04145700001845J574
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3800127133
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1105286106
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935775
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1085
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0411240001845JEAY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3837638593
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1216817509
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 941454
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 95
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0402290001835H4TE
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299425
Sectors read . . . . . *3716048446
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1088667902
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 929306
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 727
Recovers read failed . . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04137200018450CUU
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299425
Sectors read . . . . . *3682723613
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1098214375
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 933528
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1117
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0413730001845J1BW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3469637409
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1188667407
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 931093
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 586
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0413210001842K5Q3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378

```

```

Sectors read . . . . . *3536793068
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1266139183
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935981
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 828
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0412630001845J3GX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299377
Sectors read . . . . . *3188813343
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1222668486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 939603
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 22
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ038941000184509JH
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299379
Sectors read . . . . . *3233017791
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1220873045
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 931040
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 385
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04161100001841HRNK
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299379
Sectors read . . . . . *3350940077
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1172929466
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 938006
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 696
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04166600001845J555
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299370
Sectors read . . . . . *3568625156
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 985370872
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 938091
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 922
Recovers read failed . . . . . 0
Bus faults . . . . . 3

```

```

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04153300001845J517
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299398
Sectors read . . . . . *3738373803
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1002300853
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935408
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 929
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ041468000018450BYU
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1395569624
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1746883131
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1103674
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 439
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04140200001845HEUQ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299397

```

```

Sectors read . . . . . *25811233
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1252352993
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 932910
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1235
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04152400001845HSU5
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299397
Sectors read . . . . . *4123598281
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1273831207
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 932832
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 174
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1005
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ041682000018450BSG
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1036345264
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1348088619
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1109041
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 3
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 481
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0416730000184507L6
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1003555145
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1419563300
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1102888
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 726
Recovers read failed . . . . . 0
Bus faults . . . . . 1

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04143300001838HDK9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1033539280
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1491978914
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1100793
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 983
Recovers read failed . . . . . 0
Bus faults . . . . . 2

```

```

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ041406000018450CVK
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *886868524
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1418275658
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1097780
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1090
Recovers read failed . . . . . 0
Bus faults . . . . . 4

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ041411000018450BYF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299419
Sectors read . . . . . *3739073806
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1252906639
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 926683
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1657
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04165300001845JFVJ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299419

```

```

Sectors read . . . . . *3683549312
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1228174945
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 925419
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 916
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ0414920000184506ZW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299419
Sectors read . . . . . *3503891022
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1230844920
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 925241
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 2114
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Logical Drive 3 . . . . . 179106 Megabyte
Fault Tolerance . . . . . Distributed Data
Guarding
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) -22666, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04126600001845JD2V
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3308219872

```

```

Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 870817956
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1016221
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 128
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04135400001845J3JP
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299379
Sectors read . . . . . *3254707799
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 841117437
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958388
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 845
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04077500001844HM3W
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3993626463
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1396635023
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 962340
Seek errors . . . . . 0
Spin cycles . . . . . 6

```

```

Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 56
Recovers read failed . . . . . 0
Bus faults . . . . . 1

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04145700001845J574
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3800127133
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1105286106
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935951
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1085
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04112400001845JEAY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3837638593
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1216817509
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 941630
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 95
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Hard Drive 6

```

```

SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04022900001835H4TE
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299426
Sectors read . . . . . *3716048446
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1088667902
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 929482
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 727
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ041372000018450CUU
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299426
Sectors read . . . . . *3682723613
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1098214375
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 933704
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1117
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04137300001845J1BW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3469637409

```

```

Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1188667407
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 931269
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 586
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04132100001842K5Q3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299379
Sectors read . . . . . *3536793068
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1266139183
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 936157
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 828
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04126300001845J3GX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299378
Sectors read . . . . . *3188813343
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1222668486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 939779
Seek errors . . . . . 0
Spin cycles . . . . . 6

```

```

Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 22
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0389410000184509JH
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299379
Sectors read . . . . . *3233017791
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1220873045
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 931216
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 385
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04161100001841HRNK
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299379
Sectors read . . . . . *3350940077
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1172929466
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 938182
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 696
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 13

```

```

SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04166600001845J555
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299370
Sectors read . . . . . *3568625156
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 985370872
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 938267
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 922
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04153300001845J517
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299398
Sectors read . . . . . *3738373803
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1002300853
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 935584
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 929
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ041468000018450BYU
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1395569624

```

```

Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1746883131
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1103850
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 439
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04140200001845HEJQ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299397
Sectors read . . . . . *25811233
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1252352993
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 933086
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1235
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04152400001845HSU5
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299397
Sectors read . . . . . *4123598281
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1273831207
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 933008
Seek errors . . . . . 0
Spin cycles . . . . . 6

```

```

Spin up time . . . . . 174
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1005
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ041682000018450BSG
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1036345264
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1348088619
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1109217
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 481
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0416730000184507L6
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1003555145
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1419563300
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1103064
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 726
Recovers read failed . . . . . 0
Bus faults . . . . . 1

Hard Drive 20

```

```

SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04143300001838HDK9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *1033539280
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1491978914
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1100969
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 983
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ041406000018450CVK
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 341145
Sectors read . . . . . *886868524
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1418275658
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1097956
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1090
Recovers read failed . . . . . 0
Bus faults . . . . . 4

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ041411000018450BYP
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299419
Sectors read . . . . . *3739073806

```



```

Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1252906639
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 926859
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1657
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04165300001845JFJV
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299419
Sectors read . . . . . *3683549312
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1228174945
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 925595
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 916
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ0414920000184506ZW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 299419
Sectors read . . . . . *3503891022
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1230844920
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 925417
Seek errors . . . . . 0
Spin cycles . . . . . 6

```

```

Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 2114
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Drive Controller 2, Compaq Smart Array 3200 Rev. A
Controller
IDA Firmware Revision . . . . . 3.06
Array Accelerator Memory . . . . . 57344 Kbytes
Reserved for writes . . . . . 57344 Kbytes
Accelerator Status . . . . . Enabled
Battery count . . . . . 3
Batteries charged . . . . . 3
Batteries failed . . . . . 0
ProLiant . . . . . Bus 1
(internal/external), Rev. JB1F
ProLiant . . . . . Bus 2 (external),
Rev. JB1F

Logical Drive 1 . . . . . 18361 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 4395, 255, 32
Array Accelerator . . . . . Disabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04126500001845J3H8
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306304
Sectors read . . . . . *1273990089
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1258459775
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 960745
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1413
Recovers read failed . . . . . 3
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ039001000018450CW7
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *1147479872
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1142039322
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 991298
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 72
Recovers read failed . . . . . 4
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04139700001845HEJV
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *1069782310
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1281864060
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 968308
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 878
Recovers read failed . . . . . 3
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0386640000184506ZN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *854166062
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1202304643
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 960412
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 375
Recovers read failed . . . . . 2
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04134400001845J3RD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306304
Sectors read . . . . . *809062632
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1192832429
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 952234
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 691
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ041650000018330H1L
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306305
Sectors read . . . . . *779783691
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1130503131
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956275
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1483
Recovers read failed . . . . . 0

```

```

Bus faults . . . . . 3

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04038600001845JFDZ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306282
Sectors read . . . . . *867422067
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 1
Sectors written . . . . . 1245494198
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956241
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 629
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04142400001845HRBM
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 239171
Sectors read . . . . . *1493475540
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 878883701
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 932489
Seek errors . . . . . 0
Spin cycles . . . . . 10
Spin up time . . . . . 172
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 738
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0414300000184207CF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 306283
Sectors read . . . . . *882405596
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1268204109
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 953632
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 566
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04122900001845HYL3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306306
Sectors read . . . . . *379325921
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1078186180
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 957076
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 273
Recovers read failed . . . . . 1
Bus faults . . . . . 1

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04156700001845HKK0
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306306
Sectors read . . . . . *425905182
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1174329085
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 957718
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1092
Recovers read failed . . . . . 1
Bus faults . . . . . 2

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04147500001845J5FX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306306
Sectors read . . . . . *269368300
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1169021397
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958493
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 671
Recovers read failed . . . . . 1
Bus faults . . . . . 2

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04173200001845J3M2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *1102579082
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1194195431
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 967775
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 853
Recovers read failed . . . . . 0

```

```

Bus faults . . . . . 6

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ041421000018450CWD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306306
Sectors read . . . . . *897089228
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1177906218
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 965304
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1254
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04174600001845JERB
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306304
Sectors read . . . . . *843520961
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1273842952
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 961509
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 767
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04141200001845HEHN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *770216882
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1235827480
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958755
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 770
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0415360000184509Q2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *669218419
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1284844220
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 965835
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 938
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04132700001845J186
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *582283192
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1230891767
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 955900
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1202
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ03856900001838HQMD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348031
Sectors read . . . . . *2006434065
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1335916880
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1134800
Seek errors . . . . . 0
Spin cycles . . . . . 4
Spin up time . . . . . 174
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1065
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04037600001845J16M
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 330866
Sectors read . . . . . *111134250
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1080437887
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1047455
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 486
Recovers read failed . . . . . 0

```

```

Bus faults . . . . . 4

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04146500001845J161
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348033
Sectors read . . . . . *1999806158
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1220438252
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1135236
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 192
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 884
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04157300001845HEMC
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348033
Sectors read . . . . . *2264582634
Hard read errors . . . . . 1
Read errors retry . . . . . 3
ECC read errors . . . . . 0
Sectors written . . . . . 1477993507
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1133744
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 738
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0401390000H82802UF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 348033
Sectors read . . . . . *2045733196
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1501018723
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1130091
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 687
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04172100001844HNYN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *171949676
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1169000528
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 957797
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1076
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Logical Drive 2 . . . . . 13627 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 3262, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04126500001845J3H8
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306305

```

```

Sectors read . . . . . *1273990089
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1258459775
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 960921
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1413
Recovers read failed . . . . . 3
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ039001000018450CW7
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *1147479872
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1142039322
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 991474
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 72
Recovers read failed . . . . . 4
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04139700001845HEJV
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *1069782310
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1281864060
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 968484
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 878
Recovers read failed . . . . . 3
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0386640000184506ZN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *854166062
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1202304643
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 960588
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 375
Recovers read failed . . . . . 2
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04134400001845J3RD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306305
Sectors read . . . . . *809062632
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1192832429
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 952410
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 691
Recovers read failed . . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ041650000018330HLL
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306306
Sectors read . . . . . *779783691
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1130503131
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956451
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1483
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04038600001845JFDZ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306283
Sectors read . . . . . *867422067
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 1
Sectors written . . . . . 1245494198
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956417
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 629
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04142400001845HRBM
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 239172

```

```

Sectors read . . . . . *1493475540
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 878883701
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 932665
Seek errors . . . . . 0
Spin cycles . . . . . 10
Spin up time . . . . . 172
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 738
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ041430000184207CF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306284
Sectors read . . . . . *882405596
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1268204109
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 953808
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 566
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0412290001845HYL3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *379325921
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1078186180
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 957252
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 273
Recovers read failed . . . . . 1
Bus faults . . . . . 1

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0415670001845HKK0
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *425905182
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1174329085
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 957894
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1092
Recovers read failed . . . . . 1
Bus faults . . . . . 2

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ0414750001845J5FX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *269368300
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1169021397
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958669
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 671
Recovers read failed . . . . . 1
Bus faults . . . . . 2

```

```

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ0417320001845J3M2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *1102579082
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1194195431
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 967951
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 853
Recovers read failed . . . . . 0
Bus faults . . . . . 6

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04142100018450CWD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *897089228
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1177906218
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 965480
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1254
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0417460001845JERB
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306305

```

```

Sectors read . . . . . *843520961
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1273842952
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 961685
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 767
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04141200001845HEHN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *770216882
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1235827480
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958931
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 770
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0415360000184509Q2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *669218419
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1284844220
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 966011
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 938
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04132700001845J186
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *582283192
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1230891767
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956076
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1202
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ03856900001838HQMD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348032
Sectors read . . . . . *2006434065
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1335916880
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1134976
Seek errors . . . . . 0
Spin cycles . . . . . 4
Spin up time . . . . . 174
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1065
Recovers read failed . . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04037600001845J16M
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 330867
Sectors read . . . . . *111134250
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1080437887
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1047631
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 486
Recovers read failed . . . . . 0
Bus faults . . . . . 4

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04146500001845J161
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348034
Sectors read . . . . . *1999806158
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1220438252
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1135412
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 192
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 884
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04157300001845HEMC
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348034

```

```

Sectors read . . . . . *2264582634
Hard read errors . . . . . 1
Read errors retry . . . . . 3
ECC read errors . . . . . 0
Sectors written . . . . . 1477993507
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1133920
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 738
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0401390000H82802UF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348034
Sectors read . . . . . *2045733196
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1501018723
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1130267
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 687
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04172100001844HNYN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *171949676
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1169000528
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 957973
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1076
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Logical Drive 3 . . . . . 178622 Megabyte
Fault Tolerance . . . . . Distributed Data
Guarding
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) -22782, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04126500001845J3H8
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306305
Sectors read . . . . . *1273990089
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1258459775
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 961097
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1413
Recovers read failed . . . . . 3
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ039001000018450CW7
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *1147479872
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1142039322
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 991650
Seek errors . . . . . 0
Spin cycles . . . . . 6

```

```

Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 72
Recovers read failed . . . . . 4
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04139700001845HEJV
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *1069782310
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1281864060
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 968660
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 878
Recovers read failed . . . . . 3
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0386640000184506ZN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *854166062
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1202304643
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 960764
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 375
Recovers read failed . . . . . 2
Bus faults . . . . . 0

Hard Drive 5

```



```

SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04134400001845J3RD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306305
Sectors read . . . . . *809062632
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1192832429
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 952586
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 691
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ041650000018330HL1
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306306
Sectors read . . . . . *779783691
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1130503131
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956627
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1483
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04038600001845JFDZ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306283
Sectors read . . . . . *867422067

```

```

Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 1
Sectors written . . . . . 1245494198
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956593
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 629
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04142400001845HRBM
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 239172
Sectors read . . . . . *1493475540
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 878883701
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 932841
Seek errors . . . . . 0
Spin cycles . . . . . 10
Spin up time . . . . . 172
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 738
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0414300000184207CF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306284
Sectors read . . . . . *882405596
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1268204109
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 953984
Seek errors . . . . . 0
Spin cycles . . . . . 6

```

```

Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 566
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04122900001845HYL3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *379325921
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1078186180
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 957428
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 273
Recovers read failed . . . . . 1
Bus faults . . . . . 1

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04156700001845HKK0
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *425905182
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1174329085
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958070
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1092
Recovers read failed . . . . . 1
Bus faults . . . . . 2

Hard Drive 12

```

```

SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04147500001845J5FX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *269368300
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1169021397
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958845
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 671
Recovers read failed . . . . . 1
Bus faults . . . . . 2

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04173200001845J3M2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *1102579082
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1194195431
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 968127
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 853
Recovers read failed . . . . . 0
Bus faults . . . . . 6

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ041421000018450CWD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306307
Sectors read . . . . . *897089228

```

```

Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1177906218
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 965656
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1254
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04174600001845JERB
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306305
Sectors read . . . . . *843520961
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1273842952
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 961861
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 767
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04141200001845HEHN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *770216882
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1235827480
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 959107
Seek errors . . . . . 0
Spin cycles . . . . . 6

```

```

Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 770
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0415360000184509Q2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *669218419
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1284844220
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 966187
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 938
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04132700001845J186
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *582283192
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1230891767
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 956252
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1202
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19

```

```

SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ03856900001838HQMD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348032
Sectors read . . . . . *2006434065
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1335916880
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1135152
Seek errors . . . . . 0
Spin cycles . . . . . 4
Spin up time . . . . . 174
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1065
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04037600001845J16M
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 330867
Sectors read . . . . . *111134250
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1080437887
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1047807
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 486
Recovers read failed . . . . . 0
Bus faults . . . . . 4

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04146500001845J161
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348034
Sectors read . . . . . *1999806158

```

```

Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1220438252
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1135588
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 192
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 884
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04157300001845HEMC
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348034
Sectors read . . . . . *2264582634
Hard read errors . . . . . 1
Read errors retry . . . . . 3
ECC read errors . . . . . 0
Sectors written . . . . . 1477993507
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1134096
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 738
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0401390000H82802UF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 348034
Sectors read . . . . . *2045733196
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1501018723
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1130443
Seek errors . . . . . 0
Spin cycles . . . . . 3

```

```

Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 687
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04172100001844HNYN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306308
Sectors read . . . . . *171949676
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1169000528
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 958149
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1076
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Drive Controller 3, Compaq Smart Array 3200 Rev. A
Controller
IDA Firmware Revision . . . . . 3.06
Array Accelerator Memory . . . . . 57344 Kbytes
Reserved for writes . . . . . 57344 Kbytes
Accelerator Status . . . . . Enabled
Battery count . . . . . 3
Batteries charged . . . . . 3
Batteries failed . . . . . 0
ProLiant . . . . . Bus 1
(internal/external), Rev. JB1F
ProLiant . . . . . Bus 2 (external),
Rev. JB1F

Logical Drive 1 . . . . . 22025 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 5272, 255, 32
Array Accelerator . . . . . Disabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
LJ460265000019100SCE
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2871061128
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 979894723
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 632808
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 73
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ452018000019100JWW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2877649785
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 987569782
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 627381
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 96
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4610040000191000XA
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2861633405
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 976147751
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 621636
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 810
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ462059000019100LMT
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2854552916
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970545836
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 625437
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 569
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ460853000019100YZC
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2861401411
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975552892
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 618796
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 143
Recovers read failed . . . . . 0

```

```

Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4492680000191013KH
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2868544947
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 981432780
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 623135
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 2636
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ46802600002910J6JE
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2867499216
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 980006947
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 630465
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 189
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 113
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4891210000191103JL
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2865049351
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975630942
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 616066
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 65
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ491334000019091958
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2850502763
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970078208
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 612305
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 341
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ466931000019100R7Z
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2861710805
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978040187
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 616179
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 127
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJB0671600002924HQFS
Firmware Revision 1 . . . . . 3208
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 10169
Sectors read . . . . . 103346217
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 72384408
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 35026
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 39
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJB1785900002924HPV8
Firmware Revision 1 . . . . . 3208
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 10169
Sectors read . . . . . 102105068
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 71814664
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 35803
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 41
Recovers read failed . . . . . 0

```

```

Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ45732300001910099E
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2857761486
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975532543
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 624713
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 125
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ46865000002911J2V0
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2872371745
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978909575
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 620994
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 2243
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4596710000191003N2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2863563237
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973341413
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 612848
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 193
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 123
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ450919000019100B6L
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2864650714
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977672821
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 614500
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 3401
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ46152000002910J0H5
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2865567092
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978831788
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 613237
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 1705
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ460775000019091X7K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2851411024
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 971093015
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 615099
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 93
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ467346000019100YPT
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2857919589
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974854218
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 616217
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 414
Recovers read failed . . . . . 0

```

```

Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ464643000019100C5K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2868797158
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 981692345
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 620473
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 194
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 263
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ46204300001910142K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2865646711
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 979329069
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 615785
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 72
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ46759700002910J0U4
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2869534475
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 982421825
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 621244
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 162
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ43617100001910142H
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2855211857
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970403423
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 618956
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 2458
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ4854760000291H251
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2869050524
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978302303
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 618566
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 124
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Logical Drive 2 . . . . . 13627 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 3262, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
LJ460265000019100SCE
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2871061128
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 979894723
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 632984
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 73
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ452018000019100JWW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2877649785
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 987569782
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 627557
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 96
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4610040000191000XA
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2861633405
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 976147751
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 621812
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 810
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ462059000019100LMT
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2854552916
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970545836
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 625613
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 569
Recovers read failed . . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ460853000019100YZC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2861401411
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975552892
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 618972
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 143
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4492680000191013KH
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2868544947
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 981432780
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 623311
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 2636
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ46802600002910J6JE
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012

```

```

Sectors read . . . . . *2867499216
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 980006947
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 630641
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 189
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 113
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4891210000191103JL
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2865049351
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975630942
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 616242
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 65
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ491334000019091958
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2850502763
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970078208
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 612481
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 341
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ466931000019100R7Z
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2861710805
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978040187
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 616355
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 127
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJBO671600002924HQFS
Firmware Revision 1 . . . . 3208
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 10169
Sectors read . . . . . 103346217
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 72384408
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 35202
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 39
Recovers read failed . . . . 0
Bus faults . . . . . 0

```



```

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJBI785900002924HPV8
Firmware Revision 1 . . . . . 3208
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 10169
Sectors read . . . . . 102105068
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 71814664
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 35979
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 41
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ45732300001910099E
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011
Sectors read . . . . . *2857761486
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975532543
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 624889
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 125
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ46865000002911J2V0
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171011

```

```

Sectors read . . . . . *2872371745
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978909575
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 621170
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 2243
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4596710000191003N2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2863563237
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973341413
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 613024
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 193
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 123
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ450919000019100B6L
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2864650714
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977672821
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 614676
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 3401
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ46152000002910J0H5
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2865567092
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978831788
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 613413
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 1705
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ460775000019091X7K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2851411024
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 971093015
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 615275
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 93
Recovers read failed . . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ46734600019100YPT
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171012
Sectors read . . . . . *2857919589
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974854218
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 616393
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 414
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ46464300019100C5K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2868797158
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 981692345
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 620649
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 194
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 263
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ4620430001910142K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013

```

```

Sectors read . . . . . *2865646711
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 979329069
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 615961
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 72
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ4675970002910J0U4
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2869534475
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 982421825
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 621420
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 162
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4361710001910142H
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2855211857
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970403423
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 619132
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 2458
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ4854760002911H251
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 171013
Sectors read . . . . . *2869050524
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978302303
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 618742
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 124
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Drive Controller 4, Compaq Smart Array 3200 Rev. A
Controller
IDA Firmware Revision . . . . . 3.06
Array Accelerator Memory . . . . . 57344 Kbytes
Reserved for writes . . . . . 57344 Kbytes
Accelerator Status . . . . . Enabled
Battery count . . . . . 3
Batteries charged . . . . . 3
Batteries failed . . . . . 0
ProLiant . . . . . Bus 1
(internal/external), Rev. JB1F
ProLiant . . . . . Bus 2 (external),
Rev. JB1F

Logical Drive 1 . . . . . 21394 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 5121, 255, 32
Array Accelerator . . . . . Disabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ0415430001845HQRY
Firmware Revision 1 . . . . . 3207

```

```

Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 347991
Sectors read . . . . . *1021767885
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1312537482
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1148147
Seek errors . . . . . 0
Spin cycles . . . . . 2
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1132
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04130200001845J3S1
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306266
Sectors read . . . . . *3545962575
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1121778124
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 974257
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 425
Recovers read failed . . . . . 0
Bus faults . . . . . 1

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0413860000184507L3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306266
Sectors read . . . . . *3443667338
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1187047370
Hard write errors . . . . . 0

```

```

Write errors retry . . . . . 0
Seek count . . . . . 979718
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1339
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04156600001845HEMQ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306243
Sectors read . . . . . *3780655063
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1251133629
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 975865
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 632
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04136400001845HQFM
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306216
Sectors read . . . . . *3217578840
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1135728025
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 986404
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 822

```

```

Recovers read failed . . . . . 0
Bus faults . . . . . 5

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04145200001845J51U
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306216
Sectors read . . . . . *3087872887
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1135513653
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 980878
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 863
Recovers read failed . . . . . 0
Bus faults . . . . . 2

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04152900001845HR8T
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316818
Sectors read . . . . . *234856430
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1218105486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 976424
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1056
Recovers read failed . . . . . 0
Bus faults . . . . . 1

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04167100001845HRBL
Firmware Revision 1 . . . . . 3207

```

```

Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316818
Sectors read . . . . . *54170934
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1152157122
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 972614
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 956
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04174300001845JFEF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316818
Sectors read . . . . . *180514582
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1173074857
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971244
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1287
Recovers read failed . . . . . 0
Bus faults . . . . . 16

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0417240000184509QZ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306243
Sectors read . . . . . *3183980566
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1109915281
Hard write errors . . . . . 0

```

```

Write errors retry . . . . . 0
Seek count . . . . . 969896
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 2505
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04137100001845HSKP
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306244
Sectors read . . . . . *3198928918
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1209907843
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971657
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 488
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04153900001845J51W
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 347991
Sectors read . . . . . *392549753
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1339658480
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1141707
Seek errors . . . . . 0
Spin cycles . . . . . 2
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 976

```

```

Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ040024000018450BX3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306216
Sectors read . . . . . *3549884529
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1183838944
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 980430
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 89
Recovers read failed . . . . . 0
Bus faults . . . . . 1

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0402340000183703ZA
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306264
Sectors read . . . . . *4095123127
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1287685007
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971792
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 721
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ03973800001835HM9L
Firmware Revision 1 . . . . . 3207

```

```

Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306264
Sectors read . . . . . *3997652703
Hard read errors . . . . . 0
Read errors retry . . . . . 2
ECC read errors . . . . . 0
Sectors written . . . . . 1259087292
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 974969
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 74%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 132
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ041144000018450G4G
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306244
Sectors read . . . . . *3802630985
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1296812484
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 970778
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 77
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04167600001845HT5Z
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306244
Sectors read . . . . . *3654152236
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1248534662
Hard write errors . . . . . 0

```

```

Write errors retry . . . . . 0
Seek count . . . . . 973399
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 2491
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04054200001843HH9G
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306243
Sectors read . . . . . *3550498932
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1175042496
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 968865
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1268
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04135300001845J4Y6
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316816
Sectors read . . . . . *4239366637
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1280834910
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 973810
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 858

```

```

Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ04134500001845J3NC
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316816
Sectors read . . . . . *4132664388
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1243400582
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971562
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 360
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ041736000018450FX5
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306266
Sectors read . . . . . *3215606313
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1070182243
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 974870
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 1446
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04161800001845JC9A
Firmware Revision 1 . . . . . 3207

```

```

Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306267
Sectors read . . . . . *2704604299
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1109179144
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 973334
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1672
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ03999700001837HVK6
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 323434
Sectors read . . . . . *737938623
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1466818632
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1054871
Seek errors . . . . . 0
Spin cycles . . . . . 4
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 209
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04174200001845JC79
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 283048
Sectors read . . . . . *3048357617
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 948881444
Hard write errors . . . . . 0

```

```

Write errors retry . . . . . 0
Seek count . . . . . 998013
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 892
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Logical Drive 2 . . . . . 13627 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 3262, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ04154300001845HQRY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 347992
Sectors read . . . . . *1021767885
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1312537482
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1148323
Seek errors . . . . . 0
Spin cycles . . . . . 2
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1132
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ04130200001845J3S1
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306267
Sectors read . . . . . *3545962575
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1121778124
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 974433

```

```

Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 425
Recovers read failed . . . . . 0
Bus faults . . . . . 1

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0413860000184507L3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306267
Sectors read . . . . . *3443667338
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1187047370
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 979894
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1339
Recovers read failed . . . . . 0
Bus faults . . . . . 3

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04156600001845HEM0
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306244
Sectors read . . . . . *3780655063
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1251133629
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 976041
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 632
Recovers read failed . . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0413640001845HQFM
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306217
Sectors read . . . . . *3217578840
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1135728025
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 986580
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 822
Recovers read failed . . . . 0
Bus faults . . . . . 5

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0414520001845J5IU
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306217
Sectors read . . . . . *3087872887
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1135513653
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 981054
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 863
Recovers read failed . . . . 0
Bus faults . . . . . 2

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0415290001845HR8T
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes

```

```

Reference time . . . . . 316819
Sectors read . . . . . *234856430
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1218105486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 976600
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1056
Recovers read failed . . . . 0
Bus faults . . . . . 1

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0416710001845HRL
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316819
Sectors read . . . . . *54170934
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1152157122
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 972790
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . 956
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0417430001845JFEF
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316819
Sectors read . . . . . *180514582
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1173074857
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971420

```

```

Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1287
Recovers read failed . . . . 0
Bus faults . . . . . 16

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ041724000184509QZ
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306244
Sectors read . . . . . *3183980566
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1109915281
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 970072
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 2505
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0413710001845HSPK
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306245
Sectors read . . . . . *3198928918
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1209907843
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971833
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 488
Recovers read failed . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
NJ04153900001845J51W
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 347992
Sectors read . . . . . *392549753
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1339658480
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1141883
Seek errors . . . . . 0
Spin cycles . . . . . 2
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 976
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ040024000018450BX3
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306217
Sectors read . . . . . *3549884529
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1183838944
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 980606
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 89
Recovers read failed . . . . 0
Bus faults . . . . . 1

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0402340000183703ZA
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes

```

```

Reference time . . . . . 306265
Sectors read . . . . . *4095123127
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1287685007
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971968
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . 721
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ03973800001835HM9L
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306265
Sectors read . . . . . *3997652703
Hard read errors . . . . . 0
Read errors retry . . . . . 2
ECC read errors . . . . . 0
Sectors written . . . . . 1259087292
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 975145
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 74%
Seek time full . . . . . 72%
Reallocated sectors . . . . 132
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ041144000018450G4G
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306245
Sectors read . . . . . *3802630985
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1296812484
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 970954

```

```

Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 77
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ04167600001845HT5Z
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306245
Sectors read . . . . . *3654152236
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1248534662
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 973575
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . 2491
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ04054200001843HH9G
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306244
Sectors read . . . . . *3550498932
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1175042496
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 969041
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1268
Recovers read failed . . . . 0
Bus faults . . . . . 0

```



```

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0413530001845J4Y6
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316817
Sectors read . . . . . *4239366637
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1280834910
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 973986
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 858
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0413450001845J3NC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 316817
Sectors read . . . . . *4132664388
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1243400582
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 971738
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 360
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ04173600018450FX5
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes

```

```

Reference time . . . . . 306267
Sectors read . . . . . *3215606313
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1070182243
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 975046
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 1446
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0416180001845JC9A
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306267
Sectors read . . . . . *2704604299
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1109179144
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 973510
Seek errors . . . . . 0
Spin cycles . . . . . 5
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1672
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0399970001837HVK6
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 323434
Sectors read . . . . . *737938623
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1466818632
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1055047

```

```

Seek errors . . . . . 0
Spin cycles . . . . . 4
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 209
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
NJ0417420001845JC79
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 283048
Sectors read . . . . . *3048357617
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 948881444
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 998189
Seek errors . . . . . 0
Spin cycles . . . . . 3
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 892
Recovers read failed . . . . 0
Bus faults . . . . . 0

Drive Controller 5, Compaq Smart Array 3200 Rev. A
Controller
IDA Firmware Revision . . . . 3.06
Array Accelerator Memory . . . 57344 Kbytes
Reserved for writes . . . . . 57344 Kbytes
Accelerator Status . . . . . Enabled
Battery count . . . . . 3
Batteries charged . . . . . 3
Batteries failed . . . . . 0
ProLiant . . . . . Bus 1
(internal/external), Rev. JB1F
ProLiant . . . . . Bus 2 (external),
Rev. JB1F

Logical Drive 1 . . . . . 22025 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 5272, 255, 32
Array Accelerator . . . . . Disabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
LJ46663400019091MLF

```

```

Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163611
Sectors read . . . . . 1104205443
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972150509
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 631732
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 275
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ488800000191011D9
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1091840403
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 966956484
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 606163
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 153
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ480457000019101LWJ
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170760
Sectors read . . . . . *712147416
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2032678161

```

```

Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 583490
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 382
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ48977500002911J6WH
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170759
Sectors read . . . . . *711745733
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2030393110
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 584527
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 154
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ468042000019110S3D
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170759
Sectors read . . . . . *712959343
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2034709993
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582808
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%

```

```

Reallocated sectors . . . . 231
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ48289500001910H7BC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170760
Sectors read . . . . . *719155854
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2038904247
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 578490
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 74
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ462968000019100K26
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1090986815
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 966107726
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 609927
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 33
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ491781000019110YRU

```

```

Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1104260380
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973080430
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594637
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 74
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ440580000019101FJV
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166827
Sectors read . . . . . *3743814735
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1998498963
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582979
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 646
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ4580830000191005JC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166827
Sectors read . . . . . *3743251159
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1997339161

```

```

Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 567858
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 194
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 61
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ469015000019101675
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166827
Sectors read . . . . . *3746247695
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1998732388
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 567154
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 88
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJ46333300001910168W
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166827
Sectors read . . . . . *3743897258
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1996542440
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 569846
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%

```

```

Reallocated sectors . . . . 310
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ481507000019110YCU
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1089441644
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 963476593
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 605568
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 199
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 114
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ4898180000191104AC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1092681433
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 963071406
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 601426
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . 109
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ48478900002911J33E

```

```

Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1090487961
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 962172958
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 597126
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 72
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ489669000019083QR9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1080822825
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 956973534
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 596146
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 119
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ4874570000191000LD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 159814
Sectors read . . . . . 1078041310
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 948686735

```

```

Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 577145
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 151
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4896960000291H473
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1070981528
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 956142690
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595279
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 168
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ489691000019082965
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1089508425
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 962668737
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 596887
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%

```

```

Reallocated sectors . . . . 1982
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ18856300001910168M
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1093028585
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 964765294
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 597800
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 1351
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ125254000019031B68
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 195396
Sectors read . . . . . *1164713294
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 629987795
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 724819
Seek errors . . . . . 0
Spin cycles . . . . . 42
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 135
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ490027000019110S00

```

```

Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1081995241
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 961564222
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 602074
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 194
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 89
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ491415000019101LZS
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1084180449
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 961183030
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 602592
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 77
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ4749300000291H00T
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1087649156
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 961931699

```

```

Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 601558
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 225
Recovers read failed . . . . 0
Bus faults . . . . . 0

Logical Drive 2 . . . . . 13627 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 3262, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
LJ466634000019091MLF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163611
Sectors read . . . . . 1104205443
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972150509
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 631908
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 275
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ4888000000191011D9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1091840403
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 966956484
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 606339
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 153
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ480457000019101LLWJ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170760
Sectors read . . . . . *712147416
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2032678161
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 583666
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 382
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ48977500002911J6WH
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170759
Sectors read . . . . . *711745733
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2030393110
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 584703
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 154
Recovers read failed . . . . 0

```

```

Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ468042000019110S3D
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170759
Sectors read . . . . . *712959343
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2034709993
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582984
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 231
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ48289500001910H7BC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 170760
Sectors read . . . . . *719155854
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2038904247
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 578666
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 74
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ462968000019100K26
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1090986815
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 966107726
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 610103
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 33
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ491781000019110YRU
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163612
Sectors read . . . . . 1104260380
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973080430
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594813
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 74
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ440580000019101FJV
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166827
Sectors read . . . . . *3743814735
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1998498963
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 583155
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 646
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ4580830000191005JC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166828
Sectors read . . . . . *3743251159
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1997339161
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 568034
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 194
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 61
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ469015000019101675
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166828
Sectors read . . . . . *3746247695
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1998732388
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 567330
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 88
Recovers read failed . . . . 0

```

```

Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJ46333300001910168W
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 166828
Sectors read . . . . . *3743897258
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1996542440
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 570022
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . 310
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ481507000019110YCU
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1089441644
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 963476593
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 605744
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 199
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 114
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ4898180000191104AC
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1092681433
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 963071406
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 601602
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . 109
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ48478900002911J33E
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1090487961
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 962172958
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 597302
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 72
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ489669000019083QR9
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1080822825
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 956973534
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 596322
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 119
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ4874570000191000LD
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 159815
Sectors read . . . . . 1078041310
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 948686735
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 577321
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 151
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ48969600002911H473
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1070981528
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 956142690
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595455
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . 168
Recovers read failed . . . . 0

```

```

Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ489691000019082965
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1089508425
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 962668737
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 597063
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1982
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ18856300001910168M
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1093028585
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 964765294
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 597976
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1351
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ125254000019031B68
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C

```

```

Initialized for Monitoring . Yes
Reference time . . . . . 195397
Sectors read . . . . . *1164713294
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 629987795
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 724995
Seek errors . . . . . 0
Spin cycles . . . . . 42
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . 135
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ490027000019110S00
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1081995241
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 961564222
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 602250
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 194
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . 89
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ491415000019101LZS
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1084180449
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 961183030
Hard write errors . . . . . 0
Write errors retry . . . . . 0

```

```

Seek count . . . . . 602768
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 175
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 77
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ47493000002911H00T
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163613
Sectors read . . . . . 1087649156
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 961931699
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 601734
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . 225
Recovers read failed . . . . 0
Bus faults . . . . . 0

Drive Controller 6, Compaq Smart Array 3200 Rev. A
Controller
IDA Firmware Revision . . . . 3.06
Array Accelerator Memory . . . 57344 Kbytes
Reserved for writes . . . . . 57344 Kbytes
Accelerator Status . . . . . Enabled
Battery count . . . . . 3
Batteries charged . . . . . 3
Batteries failed . . . . . 0
ProLiant . . . . . Bus 1
(internal/external), Rev. JB1F
ProLiant . . . . . Bus 2 (external),
Rev. JB1F

Logical Drive 1 . . . . . 21707 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 5196, 255, 32
Array Accelerator . . . . . Disabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0

```



```

Serial Number . . . . .
LJ460289000019100GXX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1720739125
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978718290
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 613540
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 113
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ10222700001901HPJP
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1718526910
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978348576
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 608480
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 707
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ097338000019030NGW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1713781283
Hard read errors . . . . . 0
Read errors retry . . . . . 0

```

```

ECC read errors . . . . . 0
Sectors written . . . . . 973811452
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 618353
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 214
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ1234580000190314J6
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1709745697
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972702854
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 597028
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 190
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 73
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ122679000019030XC2
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1702947703
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 1
Sectors written . . . . . 968618130
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 593500
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%

```

```

Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 230
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ041311000018450GOM
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 303547
Sectors read . . . . . *4287347257
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1552048411
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1155646
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 103
Recovers read failed . . . . . 15
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ1223170000190315CP
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1722728789
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977781713
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 599090
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 115
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2

```

```

Serial Number . . . . .
LJ489952000019110YSB
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1707763453
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970178046
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 591693
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 41
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ44284400002910J83X
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1705770259
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 967436612
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 609830
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 85
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ466887000019100VBX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1718965357
Hard read errors . . . . . 0
Read errors retry . . . . . 0

```

```

ECC read errors . . . . . 0
Sectors written . . . . . 978216129
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 589156
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 155
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ466640000019100MG9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1703784287
Hard read errors . . . . . 0
Read errors retry . . . . . 3
ECC read errors . . . . . 4
Sectors written . . . . . 970253347
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594082
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 376
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJ47156400002910J9V0
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 157345
Sectors read . . . . . *1103627681
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 950478948
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 549578
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%

```

```

Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 167
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ48932600002911J2JM
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1715509367
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 976471459
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595944
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 305
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ46784000002910J6J9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1711093253
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 971642438
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594704
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 57
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2

```

```

Serial Number . . . . .
LJ46710400019101023
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1714142779
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974324748
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588097
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 412
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ467745000019100YRX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1713114147
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973850340
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 589078
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 320
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ4405210000191000KT
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163952
Sectors read . . . . . *1713593937
Hard read errors . . . . . 0
Read errors retry . . . . . 0

```

```

ECC read errors . . . . . 0
Sectors written . . . . . 975368574
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588790
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 307
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ46867100001910106L
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163952
Sectors read . . . . . *1713622852
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974865763
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 593725
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 77
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ488327000019110XD1
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163952
Sectors read . . . . . *1714306685
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975472546
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588772
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%

```

```

Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 2400
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ463724000019100MMS
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1728007795
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 983119828
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 589458
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 90
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ03158400001842JYEE
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 333948
Sectors read . . . . . *3089463311
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1469707629
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1173484
Seek errors . . . . . 0
Spin cycles . . . . . 8
Spin up time . . . . . 175
Seek time track . . . . . 0%
Seek time third . . . . . 0%
Seek time full . . . . . 0%
Reallocated sectors . . . . . 51
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4

```

```

Serial Number . . . . .
LJ491583000019050RT3
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1706962907
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972574300
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586425
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 86
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4682050000191016RE
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1705768887
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970547832
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 589240
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 141
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ467140000019101422
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1706429800
Hard read errors . . . . . 0
Read errors retry . . . . . 0

```

```

ECC read errors . . . . . 0
Sectors written . . . . . 968683021
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 587483
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 158
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Logical Drive 2 . . . . . 13627 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 3262, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
LJ460289000019100GXX
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1720739125
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978718290
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 613716
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 113
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ10222700001901HPJP
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1718526910
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978348576

```

```

Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 608656
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 707
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ09733800001903ONGW
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1713781283
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973811452
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 618529
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 214
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ1234580000190314J6
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1709745697
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972702854
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 597204
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 190
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%

```

```

Reallocated sectors . . . 73
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ122679000019030XC2
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1702947703
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 1
Sectors written . . . . . 968618130
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 593676
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . 230
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ041311000018450GOM
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 303547
Sectors read . . . . . *4287347257
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1552048411
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1155822
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 103
Recovers read failed . . . 15
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ1223170000190315CP

```

```

Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1722728789
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977781713
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 599266
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 115
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ489952000019110YSB
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1707763453
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970178046
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 591869
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . 41
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ44284400002910J83X
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1705770259
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 967436612

```

```

Hard write errors . . . . 0
Write errors retry . . . . 0
Seek count . . . . . 610006
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 85
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ466887000019100VEX
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1718965357
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978216129
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 589332
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 155
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ466640000019100MG9
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1703784287
Hard read errors . . . . . 0
Read errors retry . . . . . 3
ECC read errors . . . . . 4
Sectors written . . . . . 970253347
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594258
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%

```

```

Reallocated sectors . . . 376
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJ47156400002910J9V0
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 157345
Sectors read . . . . . *1103627681
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 950478948
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 549754
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 167
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ48932600002911J2JM
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1715509367
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 976471459
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 596120
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 305
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ46784000002910J6J9

```

```

Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1711093253
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 971642438
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594880
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 57
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ467104000019101023
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1714142779
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974324748
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588273
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 412
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ467745000019100YRX
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1713114147
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973850340

```

```

Hard write errors . . . . 0
Write errors retry . . . . 0
Seek count . . . . . 589254
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . 320
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ4405210000191000KT
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163952
Sectors read . . . . . *1713593937
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975368574
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588966
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . 307
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ46867100001910106L
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163952
Sectors read . . . . . *1713622852
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974865763
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 593901
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%

```

```

Reallocated sectors . . . 77
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ488327000019110XD1
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163952
Sectors read . . . . . *1714306685
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975472546
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588948
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . 2400
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ463724000019100MMS
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1728007795
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 983119828
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 589634
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 90
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ03158400001842JYEE

```

```

Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 333948
Sectors read . . . . . *3089463311
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 1469707629
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 1173660
Seek errors . . . . . 0
Spin cycles . . . . . 8
Spin up time . . . . . 175
Seek time track . . . . . 0%
Seek time third . . . . . 0%
Seek time full . . . . . 0%
Reallocated sectors . . . 51
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ491583000019050RT3
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163953
Sectors read . . . . . *1706962907
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972574300
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586601
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . 86
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4682050000191016RE
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163954
Sectors read . . . . . *1705768887
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970547832

```

```

Hard write errors . . . . 0
Write errors retry . . . . 0
Seek count . . . . . 589416
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 141
Recovers read failed . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ467140000019101422
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 163954
Sectors read . . . . . *1706429800
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 968683021
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 587659
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . 158
Recovers read failed . . . 0
Bus faults . . . . . 0

Drive Controller 7, Compaq Smart Array 3200 Rev. A
Controller
IDA Firmware Revision . . . . 3.06
Array Accelerator Memory . . . 57344 Kbytes
Reserved for writes . . . . 57344 Kbytes
Accelerator Status . . . . . Enabled
Battery count . . . . . 3
Batteries charged . . . . . 3
Batteries failed . . . . . 0
ProLiant . . . . . Bus 1
(internal/external), Rev. JB1F
ProLiant . . . . . Bus 2 (external),
Rev. JB1F

Logical Drive 1 . . . . . 21812 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 5221, 255, 32
Array Accelerator . . . . . Disabled

Hard Drive 1
SCSI Bus . . . . . 1

```

```

SCSI ID . . . . . 0
Serial Number . . . . .
LJ188463000019100Y5Z
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164182
Sectors read . . . . . *1745412756
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973045359
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 612472
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 189
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 486
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ457582000019100YY9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164184
Sectors read . . . . . *1753015871
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 979075408
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 608305
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 221
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4534180000191019GY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164112
Sectors read . . . . . *1749405952
Hard read errors . . . . . 0

```

```

Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977314007
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594258
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 509
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ46454000002910J806
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 150283
Sectors read . . . . . *1584663788
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 803349148
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 537021
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 92
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ457483000019100Z09
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 160040
Sectors read . . . . . *1358806800
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 958608789
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 578114
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 176

```

```

Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 225
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ467609000019100R7T
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164182
Sectors read . . . . . *1745636583
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973278486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588871
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 221
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ467526000019100YRD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164181
Sectors read . . . . . *1756800858
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 980442311
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595238
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 72
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1

```



```

SCSI ID . . . . . 2
Serial Number . . . . .
LJ481121000019101T5R
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164182
Sectors read . . . . . *1754212717
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977723832
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 596584
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 406
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ48669000002911J2MN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164183
Sectors read . . . . . *1746228279
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973833500
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 609803
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 247
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ48941300002911H40S
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164183
Sectors read . . . . . *1747240919
Hard read errors . . . . . 0

```

```

Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975234943
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 596361
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 40
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ0706980000190317LQ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164180
Sectors read . . . . . *1737260048
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970943993
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595164
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 564
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJ487291000019110XDG
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164184
Sectors read . . . . . *1740514169
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 971575909
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595316
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178

```

```

Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 222
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ49024800002911J2QY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164184
Sectors read . . . . . *1753808888
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978750655
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 599358
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 119
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ465015000019100YVF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1754757776
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978410137
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588274
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 250
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2

```

```

SCSI ID . . . . . 2
Serial Number . . . . .
LJ48677700002911J2FU
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1754873656
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975139810
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586310
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 79
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ464393000019100Y1V
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1751650162
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974319324
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586650
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 64
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ467741000019100Y5S
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1756334357
Hard read errors . . . . . 0

```

```

Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 980792582
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 583807
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 156
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4586130000191002DR
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1738495102
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973457955
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 580266
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 248
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ490058000019110X4K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1747487414
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974645451
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582568
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 191

```

```

Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 98
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ484158000019110U0S
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1758752840
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 982033580
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586440
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 45
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ467654000019100YTS
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1756122817
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978828868
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582032
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 96
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2

```

```

SCSI ID . . . . . 4
Serial Number . . . . .
LJ426262000019091YFY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1741931609
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972814876
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 579203
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 713
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ47364100001911054B
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1737902754
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 967863548
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 583792
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 189
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 96
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ467853000019100YQG
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1739487275
Hard read errors . . . . . 5

```

```

Read errors retry . . . . . 2
ECC read errors . . . . . 0
Sectors written . . . . . 968479118
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582384
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 46
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Logical Drive 2 . . . . . 13627 Megabyte
Fault Tolerance . . . . . None
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 3262, 255, 32
Array Accelerator . . . . . Enabled

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
LJ188463000019100Y5Z
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164182
Sectors read . . . . . *1745412756
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973045359
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 612648
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 189
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 486
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
LJ457582000019100YY9
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164184
Sectors read . . . . . *1753015871
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0

```

```

Sectors written . . . . . 979075408
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 608481
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 221
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ4534180000191019GY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164112
Sectors read . . . . . *1749405952
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977314007
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 594434
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 509
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ46454000002910J806
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 150283
Sectors read . . . . . *1584663788
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 803349148
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 537197
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 70%

```

```

Seek time full . . . . . 71%
Reallocated sectors . . . . . 92
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ457483000019100Z09
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 160040
Sectors read . . . . . *1358806800
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 958608789
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 578290
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 225
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ467609000019100R7T
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164182
Sectors read . . . . . *1745636583
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973278486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 589047
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 221
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 1
SCSI ID . . . . . 1

```

```

Serial Number . . . . .
LJ467526000019100YRD
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164181
Sectors read . . . . . *1756800858
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 980442311
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595414
Seek errors . . . . . 0
Spin cycles . . . . . 1
Spin up time . . . . . 180
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 72
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 8
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
LJ481121000019101T5R
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164182
Sectors read . . . . . *1754212717
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 977723832
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 596760
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 406
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
LJ48669000002911J2MN
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164183
Sectors read . . . . . *1746228279
Hard read errors . . . . . 0
Read errors retry . . . . . 0

```

```

ECC read errors . . . . . 0
Sectors written . . . . . 973833500
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 609979
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 247
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
LJ48941300002911H40S
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164183
Sectors read . . . . . *1747240919
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975234943
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 596537
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 40
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
LJ0706980000190317LQ
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164180
Sectors read . . . . . *1737260048
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 970943993
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595340
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 184
Seek time track . . . . . 33%

```

```

Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 564
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 1
SCSI ID . . . . . 6
Serial Number . . . . .
LJ487291000019110XDG
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164184
Sectors read . . . . . *1740514169
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 971575909
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 595492
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 222
Recovers read failed . . . . . 1
Bus faults . . . . . 0

Hard Drive 13
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
LJ49024800002911J2QY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164184
Sectors read . . . . . *1753808888
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978750655
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 599534
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 119
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 14
SCSI Bus . . . . . 2
SCSI ID . . . . . 1

```

```

Serial Number . . . . .
LJ465015000019100YVF
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1754757776
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978410137
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 588450
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 188
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 250
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 15
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ48677700002911J2FU
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1754873656
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 975139810
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586486
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 79
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 16
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
LJ464393000019100YIV
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1751650162
Hard read errors . . . . . 0
Read errors retry . . . . . 0

```

```

ECC read errors . . . . . 0
Sectors written . . . . . 974319324
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586826
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 183
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 64
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 17
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ467741000019100YS5
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1756334357
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 980792582
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 583983
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . . 156
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 18
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ4586130000191002DR
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1738495102
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 973457955
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 580442
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%

```

```

Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 248
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 19
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
LJ490058000019110X4K
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1747487414
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 974645451
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582744
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 191
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 98
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 20
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
LJ484158000019110U0S
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1758752840
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 982033580
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 586616
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 45
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 21
SCSI Bus . . . . . 2
SCSI ID . . . . . 3

```

```

Serial Number . . . . .
LJ467654000019100YTS
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1756122817
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 978828868
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582208
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 187
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 96
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 22
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
LJ426262000019091YFY
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1741931609
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 972814876
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 579379
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 713
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 23
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
LJ47364100001911054B
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1737902754
Hard read errors . . . . . 0
Read errors retry . . . . . 0

```

```

ECC read errors . . . . . 0
Sectors written . . . . . 967863548
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 583968
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 189
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . . 96
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Hard Drive 24
SCSI Bus . . . . . 2
SCSI ID . . . . . 6
Serial Number . . . . .
LJ467853000019100YQG
Firmware Revision 1 . . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 164185
Sectors read . . . . . *1739487275
Hard read errors . . . . . 5
Read errors retry . . . . . 2
ECC read errors . . . . . 0
Sectors written . . . . . 968479118
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 582560
Seek errors . . . . . 0
Spin cycles . . . . . 0
Spin up time . . . . . 185
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . . 46
Recovers read failed . . . . . 0
Bus faults . . . . . 0

Drive Controller 8, Compaq Smart Array 3200 Rev. A
Controller
IDA Firmware Revision . . . . . 3.06
Array Accelerator Memory . . . . . 57344 Kbytes
Reserved for writes . . . . . 57344 Kbytes
Accelerator Status . . . . . Not Configured
Battery count . . . . . 3
Batteries charged . . . . . 3
Batteries failed . . . . . 0
Internal ProLiant . . . . . Bus 1
(internal/external), Rev. JM16
Internal ProLiant . . . . . Bus 2 (external),
Rev. JM16

Logical Drive 1 . . . . . 54592 Megabyte
Fault Tolerance . . . . . Mirroring
OS Format . . . . . Multi-Sector
Distribution
Drive geometry (Cyl, Hds, Sec) 13067, 255, 32
Array Accelerator . . . . . Disabled

```

```

Hard Drive 1
SCSI Bus . . . . . 1
SCSI ID . . . . . 0
Serial Number . . . . .
NJ0416590001845J1DZ
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306474
Sectors read . . . . . *1936052474
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2664330368
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 949328
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 888
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 2
SCSI Bus . . . . . 1
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0415230001845HRBN
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 317022
Sectors read . . . . . *2698204740
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2813767512
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 941498
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 179
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 844
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 3
SCSI Bus . . . . . 1
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0413610001845J3GV
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 317024

```

```

Sectors read . . . . . *2632641087
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2703762906
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 942919
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 182
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 72%
Reallocated sectors . . . . 851
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 4
SCSI Bus . . . . . 1
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0415130001845HR4H
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306472
Sectors read . . . . . *2116714715
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2622391865
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 936623
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1417
Recovers read failed . . . . 0
Bus faults . . . . . 1

Hard Drive 5
SCSI Bus . . . . . 1
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0413590001845J18D
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306472
Sectors read . . . . . *1642969175
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2631169453
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 937579
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 176
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 857
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 6
SCSI Bus . . . . . 1
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0389620000H82500P5
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306472
Sectors read . . . . . *1584059061
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2668653199
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 938003
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 178
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 73%
Reallocated sectors . . . . 273
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 7
SCSI Bus . . . . . 2
SCSI ID . . . . . 0
Serial Number . . . . .
NJ0413330001845HRAV
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306472
Sectors read . . . . . *1887688931
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2654561936
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 944213
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 173
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 71%
Reallocated sectors . . . . 1532
Recovers read failed . . . . 0
Bus faults . . . . . 0

```

```

Hard Drive 8
SCSI Bus . . . . . 2
SCSI ID . . . . . 1
Serial Number . . . . .
NJ0411730001845JEB
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 317024
Sectors read . . . . . *3187781179
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2777393486
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 946890
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 186
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 73%
Reallocated sectors . . . . 183
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 9
SCSI Bus . . . . . 2
SCSI ID . . . . . 2
Serial Number . . . . .
NJ0403840001845JFKW
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 332282
Sectors read . . . . . *2325303043
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2880513820
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 962586
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 70%
Seek time full . . . . . 72%
Reallocated sectors . . . . 423
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 10
SCSI Bus . . . . . 2
SCSI ID . . . . . 3
Serial Number . . . . .
NJ0412620001845HRC8
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306475

```

```

Sectors read . . . . . *1866369073
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2578372230
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 937502
Seek errors . . . . . 0
Spin cycles . . . . . 7
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 61
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 11
SCSI Bus . . . . . 2
SCSI ID . . . . . 4
Serial Number . . . . .
NJ0403710001845HQDB
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 306473
Sectors read . . . . . *1527526457
Hard read errors . . . . . 22
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2609160563
Hard write errors . . . . . 0
Write errors retry . . . . . 26
Seek count . . . . . 934017
Seek errors . . . . . 0
Spin cycles . . . . . 6
Spin up time . . . . . 181
Seek time track . . . . . 33%
Seek time third . . . . . 71%
Seek time full . . . . . 71%
Reallocated sectors . . . . 521
Recovers read failed . . . . 0
Bus faults . . . . . 0

Hard Drive 12
SCSI Bus . . . . . 2
SCSI ID . . . . . 5
Serial Number . . . . .
NJ0415880001845HSU7
Firmware Revision 1 . . . . 3207
Model Number . . . . . COMPAQ HD0093172C
Initialized for Monitoring . Yes
Reference time . . . . . 317023
Sectors read . . . . . *2853802229
Hard read errors . . . . . 0
Read errors retry . . . . . 0
ECC read errors . . . . . 0
Sectors written . . . . . 2825330860
Hard write errors . . . . . 0
Write errors retry . . . . . 0
Seek count . . . . . 934210
Seek errors . . . . . 0

```

```

Spin cycles . . . . . 6
Spin up time . . . . . 177
Seek time track . . . . . 33%
Seek time third . . . . . 69%
Seek time full . . . . . 71%
Reallocated sectors . . . . 919
Recovers read failed . . . . 1
Bus faults . . . . . 1

Graphics Mode . . . . . 03 (80-Column Text)

Primary Monitor attached to . . ATI RAGE IIC PCI
Graphics Controller
with Video Graphics Color Monitor

Base Memory
System Total . . . . . 638 Kbytes
Amount Free . . . . . 596 Kbytes
(611008 Bytes)

Extended Memory
System Total . . . . . 4193280 Kbytes

Expanded Memory
LIM Driver Support . . . . . LIM driver not
loaded

Operating System . . . . . MS-DOS version 7.00
(from diskette)

Status of file: C:\IO.SYS
Date and Time . . . . . 12/11/1998
12:06:30
Size . . . . . 0 Bytes

Status of file: C:\MSDOS.SYS
Date and Time . . . . . 12/11/1998
12:06:30
Size . . . . . 0 Bytes

-----
** Dump of C:\CONFIG.SYS
-----
** End of file

Status of file: C:\COMMAND.COM
*** File does not exist

-----
** Dump of C:\AUTOEXEC.BAT
-----
** End of file

Environment variables
PATH=
PROMPT=$P$G
COMSPEC=A:\COMMAND.COM
CMDLINE=inspect /u

```



```

End of environment

Revisions Table

Previous Revisions
I/O Board Revision . . . . . 01
  Assembly Version . . . . . 1
  Functional Revision Level . . . . . A
Memory Board 1 Revision . . . . . 01
  Assembly Version . . . . . 1
  Functional Revision Level . . . . . A
Power backplane revision . . . . . 01
  Assembly Version . . . . . 1
  Functional Revision Level . . . . . A

Current Revisions
I/O Board Revision . . . . . 01
  Assembly Version . . . . . 1
  Functional Revision Level . . . . . A
Memory Board 1 Revision . . . . . 01
  Assembly Version . . . . . 1
  Functional Revision Level . . . . . A
Power backplane revision . . . . . 01
  Assembly Version . . . . . 1
  Functional Revision Level . . . . . A

System Configuration Utility . . Version 2.43

Extended Non-volatile Memory
-----
D:\DIRECT.OUT 0
Slot Type . . . . . Embedded
Board ID . . . . . CPQ1601
CFG File Extension
Revision Level . . . . . 2.21
Type Entry(s) . . . . . MSD,FPYCTL
  IRQ Entry(s):
    IRQ 6, Not Shared, Edge Triggered
  DMA Channel(s):
    Channel 2, Not Shared
    Timing: Type B
    Transfer Size: 8-bit (byte)
  Port Range(s):
    03F0h - 03F5h, Not Shared
    03F6h - 03F7h, Shared

  Type Entry(s) . . . . .
MSD,UNIT0,FPYDRV;TYP=4
  Type Entry(s) . . . . .
MSD,UNIT1,FPYDRV;TYP=0
  Type Entry(s) . . . . . MSD
  Type Entry(s) . . . . . MSD,DSKCTL;CTL2
  IRQ Entry(s):
    IRQ 14, Not Shared, Edge Triggered
  Port Range(s):
    01F0h - 01F7h, Not Shared
    03F6h, Shared
    11F1h, Not Shared

  Type Entry(s) . . . . . MSD,UNIT0,DSKDRV;CD
  Type Entry(s) . . . . . MSD,UNIT1,DSKDRV
Memory Entry(s):

```

```

Size Range
-----
ROM: Other, Cacheable 896K - 1M
128K
  Type Entry(s) . . . . . MEM;COMPAQ
  Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 0K - 640K
640K
  Type Entry(s) . . . . . MEM;COMPAQ
  Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 1M - 16M
15M
  Type Entry(s) . . . . . MEM;COMPAQ
  Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 16M - 64M
48M
RAM: System, Cacheable 64M - 128M
64M
RAM: System, Cacheable 128M - 192M
64M
RAM: System, Cacheable 192M - 256M
64M
RAM: System, Cacheable 256M - 320M
64M
RAM: System, Cacheable 320M - 384M
64M
RAM: System, Cacheable 384M - 448M
64M
RAM: System, Cacheable 448M - 512M
64M
  Type Entry(s) . . . . . MEM;COMPAQ
  Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 512M - 576M
64M
RAM: System, Cacheable 576M - 640M
64M
RAM: System, Cacheable 640M - 704M
64M
RAM: System, Cacheable 704M - 768M
64M

```

```

RAM: System, Cacheable 768M - 832M
64M
RAM: System, Cacheable 832M - 896M
64M
RAM: System, Cacheable 896M - 960M
64M
RAM: System, Cacheable 960M - 1024M
64M
  Type Entry(s) . . . . . MEM;COMPAQ
  Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 1024M - 1088M
64M
RAM: System, Cacheable 1088M - 1152M
64M
RAM: System, Cacheable 1152M - 1216M
64M
RAM: System, Cacheable 1216M - 1280M
64M
RAM: System, Cacheable 1280M - 1344M
64M
RAM: System, Cacheable 1344M - 1408M
64M
RAM: System, Cacheable 1408M - 1472M
64M
RAM: System, Cacheable 1472M - 1536M
64M
  Type Entry(s) . . . . . MEM;COMPAQ
  Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 1536M - 1600M
64M
RAM: System, Cacheable 1600M - 1664M
64M
RAM: System, Cacheable 1664M - 1728M
64M
RAM: System, Cacheable 1728M - 1792M
64M
RAM: System, Cacheable 1792M - 1856M
64M
RAM: System, Cacheable 1856M - 1920M
64M
RAM: System, Cacheable 1920M - 1984M
64M
RAM: System, Cacheable 1984M - 2048M
64M
  Type Entry(s) . . . . . MEM;COMPAQ
  Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 2048M - 2112M
64M

```

```

RAM: System, Cacheable 2112M - 2176M
64M RAM: System, Cacheable 2176M - 2240M
64M RAM: System, Cacheable 2240M - 2304M
64M RAM: System, Cacheable 2304M - 2368M
64M RAM: System, Cacheable 2368M - 2432M
64M RAM: System, Cacheable 2432M - 2496M
64M RAM: System, Cacheable 2496M - 2560M
64M
Type Entry(s) . . . . . MEM;COMPAQ
Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 2560M - 2624M
64M RAM: System, Cacheable 2624M - 2688M
64M RAM: System, Cacheable 2688M - 2752M
64M RAM: System, Cacheable 2752M - 2816M
64M RAM: System, Cacheable 2816M - 2880M
64M RAM: System, Cacheable 2880M - 2944M
64M RAM: System, Cacheable 2944M - 3008M
64M RAM: System, Cacheable 3008M - 3072M
64M
Type Entry(s) . . . . . MEM;COMPAQ
Memory Entry(s):
Size Range
-----
RAM: System, Cacheable 3072M - 3136M
64M RAM: System, Cacheable 3136M - 3200M
64M RAM: System, Cacheable 3200M - 3264M
64M RAM: System, Cacheable 3264M - 3328M
64M RAM: System, Cacheable 3328M - 3392M
64M RAM: System, Cacheable 3392M - 3456M
64M RAM: System, Cacheable 3456M - 3520M
64M RAM: System, Cacheable 3520M - 3584M
64M
Type Entry(s) . . . . . MEM;COMPAQ
Memory Entry(s):

```

```

Size Range
-----
RAM: System, Cacheable 3584M - 3648M
64M RAM: System, Cacheable 3648M - 3712M
64M RAM: System, Cacheable 3712M - 3776M
64M RAM: System, Cacheable 3776M - 3840M
64M RAM: System, Cacheable 3840M - 3904M
64M RAM: System, Cacheable 3904M - 3936M
32M
Type Entry(s) . . . . . MEM;COMPAQ
Type Entry(s) . . . . . MEM;COMPAQ
Type Entry(s) . . . . . COM,ASY;COM1;A
IRQ Entry(s):
IRQ 4, Not Shared, Edge Triggered
Port Range(s):
03F8h - 03FFh, Not Shared
Type Entry(s) . . . . . COM,ASY;COM2;B
IRQ Entry(s):
IRQ 3, Not Shared, Edge Triggered
Port Range(s):
02F8h - 02FFh, Not Shared
Type Entry(s) . . . . . PAR;LPT1
IRQ Entry(s):
IRQ 7, Not Shared, Edge Triggered
DMA Channel(s):
Channel 0, Not Shared
Timing: ISA Compatible
Transfer Size: 8-bit (byte)
Port Range(s):
03BCh - 03BEh, Not Shared
Type Entry(s) . . . . . PTR,8042
IRQ Entry(s):
IRQ 12, Not Shared, Edge Triggered
IRQ Entry(s):
IRQ 13, Shared, Edge Triggered
Type Entry(s) . . . . . OTH,A20
Type Entry(s) . . . . . OTH,SOFTNMI
Type Entry(s) . . . . . OTH,FLSFNMI
Type Entry(s) . . . . . OTH,BUSNMI
Type Entry(s) . . . . . OTH,DSKTDMA
Type Entry(s) . . . . . OTH,REFRESH
Type Entry(s) . . . . . OTH,PERR
Type Entry(s) . . . . . OTH,HPK
Type Entry(s) . . . . . OTH,ACPI
IRQ Entry(s):
IRQ 9, Not Shared, Level Triggered
Type Entry(s) . . . . . OTH,WARMBOOT
Type Entry(s) . . . . . OTH,IOQ
Type Entry(s) . . . . . OTH,F1BP
Type Entry(s) . . . . . OTH,TABLE;FULLTABLE
IRQ Entry(s):
IRQ 13, Shared, Edge Triggered

```

```

Type Entry(s) . . . . . ISA;MAP
Free Form Text . . . . . 21 82 83 84 85 86
87 88 89 8A 8B 8C E0 E0
E0 00 00 00 00 00
00 00 00 00 00 00 00 00
00 00
Type Entry(s) . . . . . ISA;PCIMAP
Free Form Text . . . . . 03 00 08 04 00 10
05 04 08 06 04 10 07 08
08 08 08 10 09 08
18 0A 08 20 0B 08 28
D:\DIRECT.OUT 2
Slot Type . . . . . Expansion
Board ID . . . . . CPQ1603
CFG File Extension
Revision Level . . . . . 2.21
Memory Allocation (including INSPECT)
PSP SIZE NAME TRAPPED INTERRUPTS
-----
088F 007200 COMMAND.COM 2Fh 2Eh 24h 23h 22h
0A5A 197856 INSPECT.EXE DDh FFh F9h F4h F3h
F2h F1h EFh
E5h DEh 3Fh 00h
System Configuration Memory
00 - 0F : 46 00 17 00 17 00 03 03 02 99 26
82 50 80 00 00
10 - 1F : 40 00 00 00 03 80 02 00 3C 00 00
00 00 00 00 02
20 - 2F : 00 00 00 00 7F 20 20 40 00 92 00
00 00 10 02 A4
30 - 3F : 00 3C 19 80 00 00 XX XX XX XX XX XX
XX XX XX XX XX
BIOS Data Area
40:0000 : F8 03 F8 02 00 00 00 00 BC 03 00
00 00 00 80 9F
40:0010 : 27 44 00 7E 02 00 00 00 00 00 1E
00 1E 00 00 00
40:0020 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:0030 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 01 01
40:0040 : 24 00 04 00 00 2E 01 0B 02 03 50
00 00 10 00 00
40:0050 : 00 14 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:0060 : 0E 0D 00 D4 03 29 30 C2 11 05 77
00 CB 49 11 00
40:0070 : 00 00 00 12 00 01 00 00 14 14 14
14 01 01 01 01
40:0080 : 1E 00 3E 00 18 10 00 60 F9 11 0B
01 00 00 00 05
40:0090 : 17 00 00 00 2E 00 10 00 00 00 00
00 00 00 00 00
40:00A0 : 00 00 00 00 00 00 00 00 7A 13 00
C0 00 00 00 00
40:00B0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:00C0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00

```

```

40:00D0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:00E0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
40:00F0 : 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00

```

Interrupt Vector Table (including INSPECT)

```

00 - 03 : 0A6A:0555 0070:0465
07C6:0016 0070:0465
04 - 07 : 0070:0465 F000:FF54
F000:93CC F000:9BD0
08 - 0B : 07C6:001F 07C6:0028
F000:9BD0 F000:9BD0
0C - 0F : F000:9BD0 F000:9BD0
07C6:009A 0070:0465
10 - 13 : C000:12FC F000:F84D
F000:F841 0070:03EE
14 - 17 : F000:DDA3 0206:0240
0070:042D F000:EPD2
18 - 1B : F000:F06D 0887:002F
F000:FE6E 0070:045F
1C - 1F : F000:FF53 F000:0000
0000:0522 C000:2037
20 - 23 : 00C9:0FA8 00C9:0FB2
088F:0314 088F:016D
24 - 27 : 088F:0178 00C9:0FBC
00C9:0FC6 00C9:0FD0
28 - 2B : 00C9:106F 0070:0466
00C9:106F 00C9:106F
2C - 2F : 00C9:106F 00C9:106F
088F:0162 0890:01CC
30 - 33 : C90F:E4EA F000:9B00
00C9:106F 00C9:106F
34 - 37 : 00C9:106F 00C9:106F
00C9:106F 00C9:106F
38 - 3B : 00C9:106F 00C9:106F
00C9:106F 00C9:106F
3C - 3F : 00C9:106F 00C9:106F
00C9:106F 1B76:04F7
40 - 43 : F000:EC59 0000:0000
F000:F065 C000:244A
44 - 47 : F000:9BD0 F000:9BD0
0000:0000 F000:9BD0
48 - 4B : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
4C - 4F : F000:9BD0 F000:9BD0
F000:9BD0 0070:04FC
50 - 53 : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
54 - 57 : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
58 - 5B : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
5C - 5F : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0
60 - 63 : 0000:0000 0000:0000
0000:0000 0000:0000
64 - 67 : 0000:0000 0000:0000
0000:0000 0000:0000
68 - 6B : F000:9BD0 F000:9BD0
F000:9BD0 F000:9BD0

```

```

6C - 6F : F000:9BD0 C000:12FC
F000:9BD0 F000:9BD0
70 - 73 : 07C6:0035 F000:9C1F
F000:9BD0 07C6:00CA
74 - 77 : 07C6:00E2 F000:9C28
07C6:00FA F000:9BD0
78 - 7B : 0000:0000 0000:0000
0000:0000 0000:0000
7C - 7F : 0000:0000 0000:0000
0000:0000 0000:0000
80 - 83 : 0000:0000 0000:0000
0000:0000 0000:0000
84 - 87 : 0000:0000 0000:0000
0000:0000 0000:0000
88 - 8B : 0000:0000 0000:0000
0000:0000 0000:0000
8C - 8F : 0000:0000 0000:0000
0000:0000 0000:0000
90 - 93 : 0000:0000 0000:0000
0000:0000 0000:0000
94 - 97 : 0000:0000 0000:0000
0000:0000 0000:0000
98 - 9B : 0000:0000 0000:0000
0000:0000 0000:0000
9C - 9F : 0000:0000 0000:0000
0000:0000 0000:0000
A0 - A3 : 0000:0000 0000:0000
0000:0000 0000:0000
A4 - A7 : 0000:0000 0000:0000
0000:0000 0000:0000
A8 - AB : 0000:0000 0000:0000
0000:0000 0000:0000
AC - AF : 0000:0000 0000:0000
0000:0000 0000:0000
B0 - B3 : 0000:0000 0000:0000
0000:0000 0000:0000
B4 - B7 : 0000:0000 0000:0000
0000:0000 0000:0000
B8 - BB : 0000:0000 0000:0000
0000:0000 0000:0000
BC - BF : 0000:0000 0000:0000
0000:0000 0000:0000
C0 - C3 : 0000:0000 0000:0000
0000:0000 0000:0000
C4 - C7 : 0000:0000 0000:0000
0000:0000 0000:0000
C8 - CB : 0000:0000 0000:0000
0000:0000 0000:0000
CC - CF : 0000:0000 0000:0000
0000:0000 0000:0000
D0 - D3 : 0000:0000 0000:0000
0000:0000 0000:0000
D4 - D7 : 0000:0000 0000:0000
0000:0000 0000:0000
D8 - DB : 0000:0000 0000:0000
0000:0000 0000:0000
DC - DF : 0000:0000 9394:0000
0093:F000 0200:08DA
E0 - E3 : A700:0030 0206:0058
A6DC:0950 03E8:2051
E4 - E7 : 00A4:0080 0019:A6D0
9ABF:80D8 6B70:1000

```

```

E8 - EB : F000:A6C9 6B70:0000
0292:1000 6B5D:0001
EC - EF : 0002:081F 0006:5E9F
0046:0046 081C:FF00
F0 - F3 : 0046:5FC4 1D21:1707
1707:5ECA 1400:0801
F4 - F7 : 1BCE:0246 0101:7266
0000:0000 0000:601C
F8 - FB : 601C:0020 14B5:6322
00CA:0003 0000:061A
FC - FF : 54BD:72C3 0000:00F6
7D3A:0000 0246:F000

```

PCI Devices Information

```

Signature . . . . . PCI
Config Mechanism #1 . . . . . Supported
Config Mechanism #2 . . . . . Not Supported
Spec Cycle for Config #1 . . . . . Supported
Spec Cycle for Config #2 . . . . . Not Supported
BIOS Interface Version . . . . . 2.10
Last PCI Bus Number . . . . . 14
Number of PCI Devices . . . . . 13

Bus Number . . . . . 0
Device Number . . . . . 13
Function Number . . . . . 00h
Slot Number . . . . . 0
Vendor ID . . . . . 1000h
Device ID . . . . . 000Fh
Revision ID . . . . . 04h
Device Type . . . . . SCSI Bus Controller
Programming Interface . . . . . 00h
Expansion ROM Base Address . . . . . 0h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#
IO Address Base . . . . . 2800h
IO Address Length . . . . . 100h
Memory Address Base . . . . . F73C0000h
Memory Address Length . . . . . 100h
Memory Address Base . . . . . F73B0000h
Memory Address Length . . . . . 1000h

Bus Number . . . . . 0
Device Number . . . . . 14
Function Number . . . . . 00h
Slot Number . . . . . 0
Vendor ID . . . . . 1002h
Device ID . . . . . 4756h
Revision ID . . . . . 7Ah
Device Type . . . . . VGA Compatible
Controller
Programming Interface . . . . . 00h
Expansion ROM Base Address . . . . . FFFE0000h
IRQ Line . . . . . 255
IRQ Pin . . . . . Not Used
Memory Address Base . . . . . F6000000h
Memory Address Length . . . . . 1000000h
IO Address Base . . . . . 2C00h
IO Address Length . . . . . 100h
Memory Address Base . . . . . F73A0000h
Memory Address Length . . . . . 1000h

Bus Number . . . . . 1

```

```

Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 3
Vendor ID . . . . . 0E11h
Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 15
IRQ Pin . . . . . INTA#

Bus Number . . . . . 2
Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 4
Vendor ID . . . . . 0E11h
Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#

Bus Number . . . . . 4
Device Number . . . . . 9
Function Number . . . . . 00h
Slot Number . . . . . 0
Vendor ID . . . . . 0E11h
Device ID . . . . . 000Fh
Revision ID . . . . . 14h
Device Type . . . . . SCSI Bus Controller
Programming Interface . . . . 00h
Expansion ROM Base Address . . 0h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#
IO Address Base . . . . . 6000h
IO Address Length . . . . . 100h
Memory Address Base . . . . . F76F0000h
Memory Address Length . . . . 100h
Memory Address Base . . . . . F76E0000h
Memory Address Length . . . . 1000h

Bus Number . . . . . 4
Device Number . . . . . 9
Function Number . . . . . 01h
Slot Number . . . . . 0
Vendor ID . . . . . 0E11h
Device ID . . . . . 000Fh
Revision ID . . . . . 14h
Device Type . . . . . SCSI Bus Controller
Programming Interface . . . . 00h
Expansion ROM Base Address . . 0h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTB#
IO Address Base . . . . . 6400h
IO Address Length . . . . . 100h
Memory Address Base . . . . . F76D0000h
Memory Address Length . . . . 100h
Memory Address Base . . . . . F76C0000h
Memory Address Length . . . . 1000h

```

```

Bus Number . . . . . 5
Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 5
Vendor ID . . . . . 0E11h
Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#

Bus Number . . . . . 6
Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 6
Vendor ID . . . . . 0E11h
Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#

Bus Number . . . . . 8
Device Number . . . . . 4
Function Number . . . . . 00h
Slot Number . . . . . 10
Vendor ID . . . . . 0E11h
Device ID . . . . . 1229h
Revision ID . . . . . 05h
Device Type . . . . . Ethernet Controller
Programming Interface . . . . 00h
Expansion ROM Base Address . . FFF00000h
IRQ Line . . . . . 5
IRQ Pin . . . . . INTA#
Memory Address Base . . . . . F79F0000h
Memory Address Length . . . . 1000h
IO Address Base . . . . . 9000h
IO Address Length . . . . . 20h
Memory Address Base . . . . . F7B00000h
Memory Address Length . . . . 100000h

Bus Number . . . . . 9
Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 7
Vendor ID . . . . . 0E11h
Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#

Bus Number . . . . . 10
Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 8
Vendor ID . . . . . 0E11h

```

```

Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#

Bus Number . . . . . 11
Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 9
Vendor ID . . . . . 0E11h
Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#

Bus Number . . . . . 12
Device Number . . . . . 0
Function Number . . . . . 00h
Slot Number . . . . . 11
Vendor ID . . . . . 0E11h
Device ID . . . . . AE10h
Revision ID . . . . . 03h
Device Type . . . . . Other Mass Storage
Controller
Programming Interface . . . . 00h
IRQ Line . . . . . 11
IRQ Pin . . . . . INTA#

```

PROLIANT 7000 is a trademark of Compaq Computer Corporation.

Appendix D: 180-Day Space

Warehouses	1830	tpmc	22478.9	tpmCW	12.28	
Table	Rows	Data	Index	5% Space	8% Space	Total Space in meg
Warehouse	1,834	200	40	10		0.20
District	18,300	2,040	64	102		1.95
Item	100,000	9,528	88	476		9.30
New-order	16,470,000	260,400	736		36,600	290.04
History	54,900,000	3,050,064	0		599,450	3,563.5
Orders	54,900,000	1,682,760	929,488		513,403	2,144.6
Customer	54,900,000	39,927,280	2,563,848	1,996,364		38,991.4
Order-line	549,000,000	34,312,920	85,480		6,578,018	39,932.5
Stock	183,000,000	58,560,000	131,248	2,928,000		57,187.5
Totals		137,805,182	3,710,992	4,924,952	7,727,470	142,122
File Groups	Files	Sizes	InMeg	Needed in meg	Overhead	Not Needed
mic-fg	7	12288000	84,000	45,943	459	37,596
big-fg	7	17408000	119,000	96,179	48	22,773
	0	0	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
Totals			203,000	142,122	508	60,371
Dynamic space	39,045,745	sum of Data For Order, Order-Line and History (excluding Free e				
Static space	107,395,900	Data + Index + 5% Space + Overhead - Dynamic space				
Free space	(389,284)	Total Seg. Size - Dynamic Space - Static Space - Not Needed				
Daily growth	7,673,927	Dynamic space/W * 62.5) * tpmC				
Daily spread	(11,900,174)	Free space - 1.5 * Daily growth (zero if negative)				
180 day (KB)	1,488,702,738	Static space + 180 (daily growth + daily spread)				
180 day (GB)	1419.74	Excludes OS, Paging and RWMS Logs				
Log size (MB)	16999992	Total size of log file				
%Log used	3.000%	of log file used during entire run				
Total N-O'n	91466	Total count of N-O transactions during entire run				
Log per N-O'n	5,847	Number of KB per New-Order transaction				
8 Hour log (GB)	58.75					
Mirrored=	118					
Configured disks						
Data disks =	# of Disks	'd size				
	0	3.99	0			
	168	8.68	1,458			
			1,458			
Log disks =	# of Disks	'd size				
	0	3.99	0			
	14	8.68	122			
Priced disks						
Data disks =	# of Disks	'd size				
	0	3.99	0			
	168	8.68	1,458	Need	1,420	
Log disks =	# of Disks	'd size				
	0	3.99	0			
COMPACT DISK UTILIZATION REPORT						
Need 118						

Appendix E:

Third Party Letters

FEB 05 1999 14:03 FR MICROSOFT_RECV #1 425 936 7329 TO 912815148375 P.03/03
One Microsoft Way
Redmond, WA 98052-6399
Fax 425 936 7329
<http://www.microsoft.com/>

Microsoft

February 5, 1999

Mr. Michael Engbrock
Compaq Computer Corporation
P. O. Box 692000, MS 090803
Houston, TX 77269-2000
via FAX: (281) 514 - 8375

Dear Mike,

Here is the information you requested regarding US pricing of certain Microsoft products:

Microsoft SQL Server 7.0, Enterprise Edition, unlimited user license	\$28999
Microsoft Windows NT Server 4.0, Enterprise Edition, incl 25 CALs	\$3999
Windows NT Server 4.0, incl 5 CALs	\$809
Visual C++ Professional 5.0	\$499
5-yr maintenance for above software @ \$2095/yr	\$10475

This quote is valid for the next 60 days. Please let me know if I can be of any further assistance.

Best regards,



Sid Arora
Product Manager, Microsoft SQL Server
Applications Marketing

Microsoft Corporation is an equal opportunity employer.

** TOTAL PAGE.03 **

FEB 05 1999 14:03 FR MICROSOFT RECP #1 425 936 7329 TO 912815148375 P. 02/03
One Microsoft Way Fax 425 936 7329
Redmond, WA 98052-6399 <http://www.microsoft.com/>

Microsoft

February 5, 1999

Mr. Mike Engbrock
Compaq Computer Corporation
P.O. Box 692000
Houston, TX 77269-2000
via FAX: (281) 514-8375


Dear Mike,

Microsoft has received your request for permission to disclose the results of TPC-C benchmark tests conducted by Compaq with Microsoft SQL Server 7.0 Enterprise Edition on the following system:

Compaq ProLiant 7000 server, 4-processors, Pentium II Xeon, 450 MHz
Test results: 22000 ipmC @ \$21/ipmC approximately

Microsoft hereby grants Compaq permission to disclose these results to third parties and acknowledges that Compaq has formally requested permission to do so in accordance with the license agreement for Microsoft SQL Server 7.0 software.

Best regards,


Sid Arora
Product Manager, Microsoft SQL Server
Applications Marketing

Microsoft Corporation is an equal opportunity employer.

ALLSTAR SYSTEMS

Item Description	Stock Number	Price Each	QTY	Total	Service	Total Price
ProLant 7000-6450-2S	388671-001	32,928.00	1	\$32,928.00		\$32,928.00
ProLant Warranty Upgrade	276535-000	1,120.00	1		\$1,120.00	\$1,120.00
Onsite 5x8 & Hr Response		0.00	0		\$11,962.00	\$11,962.00
Maintenance						
ProLant Xeon 450 - 2MB Processor Kit	328814-B21	4,816.00	2	\$9,632.00		\$9,632.00
512MB EDO DIMM Memory Kit (Sons)	328582-B21	2,004.80	7	\$14,033.60		\$14,033.60
Compaq SMART Array Controller 3200	295643-B21	1,960.00	7	\$13,720.00		\$13,720.00
ProLant Storage System -UE	348710-002	1,902.88	14	\$26,640.32		\$26,640.32
ProLant Storage Warranty Upgrade	276537-001	430.00	14		\$6,020.00	\$6,020.00
4.3-GB LW SCSI-3 10,000 RPM Drive	304659-B21	502.88	1	\$502.88		\$502.88
9.1 GB Hot-Plug SCSI-3 Drive - 10K	336357-B21	638.58	182	\$171,021.76		\$171,021.76
Compaq VGA V55 Color Monitor	306006-001	228.58	5	\$1,198.40		\$1,198.40
48GB DAT Drive	295353-B21	703.36	1	\$703.36		\$703.36
Compaq UPS Model T2000	242688-005	1,786.40	1	\$1,786.40		\$1,786.40
ProLant 1800 8/400 - 64MB, 100TX Nic, 4.3GB	315570-001	2,486.48	4	\$9,985.92		\$9,985.92
ProLant Warranty Upgrade	276535-000	1,120.00	4		\$4,480.00	\$4,480.00
Onsite 5x8 4 Hr Response		0.00	0		\$2,378.34	\$2,378.34
Maintenance						
ProLant 1600 6400 Processor Option Kit	313612-B21	1,064.00	4	\$4,256.00		\$4,256.00
64-Megabyte SD DIMM Kit	313614-B21	225.12	4	\$900.48		\$900.48
128-Megabyte SD DIMM Kit	313615-B21	464.80	8	\$3,718.40		\$3,718.40
Compaq SW3822 24Port Dual Spd Switch	253300-B21	2,419.20	4	\$9,676.80		\$9,676.80
E-Port 10BaseT hub		29.00	2550	\$72,950		\$72,950.00
MKS Toolkit for NT	0	499.00	1	\$499.00		\$499.00
Large Purchase and Cash discount				(\$33,558)		(\$33,558.11)
Totals:				\$341,595.21	\$26,950.35	\$367,555.55

Please include a large volume and cash price discount and may vary when items are purchased separately.
 Service prices based upon a service contract for all items listed.
 On-site 5 days a hour day / 4 hour response upgrade to Compaq 2nd day Service
 The above quotation is valid for at least 60 days.
 Compaq Warranty Extension upgrade the standard 3 year parts warranty to years 4 and 5.

Houston Office: 6401 Southwest Freeway, Houston, Texas 77074; (713) 795-5000 • Dallas Office: 14802 Preston Road, Dallas, Texas 75244; (214) 774-7000