



TPC Benchmark™ C Full Disclosure Report

**Unisys e-@ction Enterprise Server
ES7000**

using

Microsoft SQL Server 2000 Enterprise Edition

on

Microsoft Windows Datacenter Server

Limited Edition

**First Edition
September 10th 2001**

Unisys Part Number 4500 5204-000

First Edition – September 2001

Unisys Corporation 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. Unisys Corporation 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, Unisys Corporation and Microsoft Corporation provide no warranty on the pricing information in this document.

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 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 was obtained in a rigorously controlled environment, and therefore results obtained in other operating environments may vary significantly. Unisys Corporation and Microsoft Corporation do 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 © 2001 Unisys 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 on the title page of each item reproduced.

Printed in USA, September 2001.

Unisys Corporation Part Number: 4500 5204-000

Unisys and e-@ction are registered trademarks of Unisys Corporation.

Intel, Pentium III and Xeon are registered trademarks of Intel Corporation.

Microsoft, Windows, Visual C++ and SQL Server 2000 are registered trademarks of Microsoft Corporation.

cLAN is a registered trademark of Emulex Corporation.

TPC Benchmark, TPC-C and tpmC are trademarks of the Transaction Processing Performance Council.

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

Page	Issue
i through xii	-000
0-1 through 0-4	-000
1-1 through 1-1	-000
1-2	Blank
2-1 through 2-2	-000
3-1 through 3-3	-000
3-4	Blank
4-1 through 4-6	-000
5-1 through 5-8	-000
6-1 through 6-2	-000
7-1 through 7-2	-000
8-1 through 8-1	-000
8-2	Blank
9-1 through 9-3	-000
9-4	Blank
A-1 through A-42	-000
B-1 through B-57	-000
B-58	Blank
C-1 through C-62	-000
D-1 through D-11	-000
D-12	Blank
E-1 through E-2	-000
F-1 through F-3	-000
F-4	Blank

Unisys uses an 11-digit document numbering system. The suffix of the document number (1234 5678-xyz) indicates the document level. The first digit of the suffix (x) designates a revision level; the second digit (y) designates an update level. For example, the first release of a document has a suffix of -000. A suffix of -130 designates the third update to revision 1. The third digit (z) is used to indicate an errata for a particular level and is not reflected in the page status summary.

Overview

This report documents the methodology and results of the TPC Benchmark C (TPC-C) conducted on the Unisys Corporation e-@ction Enterprise Server ES7000. The operating system on the server was Microsoft Windows Datacenter Server Limited Edition. The DBMS used was Microsoft SQL Server 2000 Enterprise Edition. The operating system on the clients was Microsoft Windows 2000 Server. The clients ran Microsoft's Internet Information Server 5.0 and COM+.

TPC Benchmark Metrics

The standard TPC Benchmark C metrics, tpmC (transactions per minute), price per tpmC (three year capital cost per measured tpmC), and the availability date are reported as required by the benchmark specification.

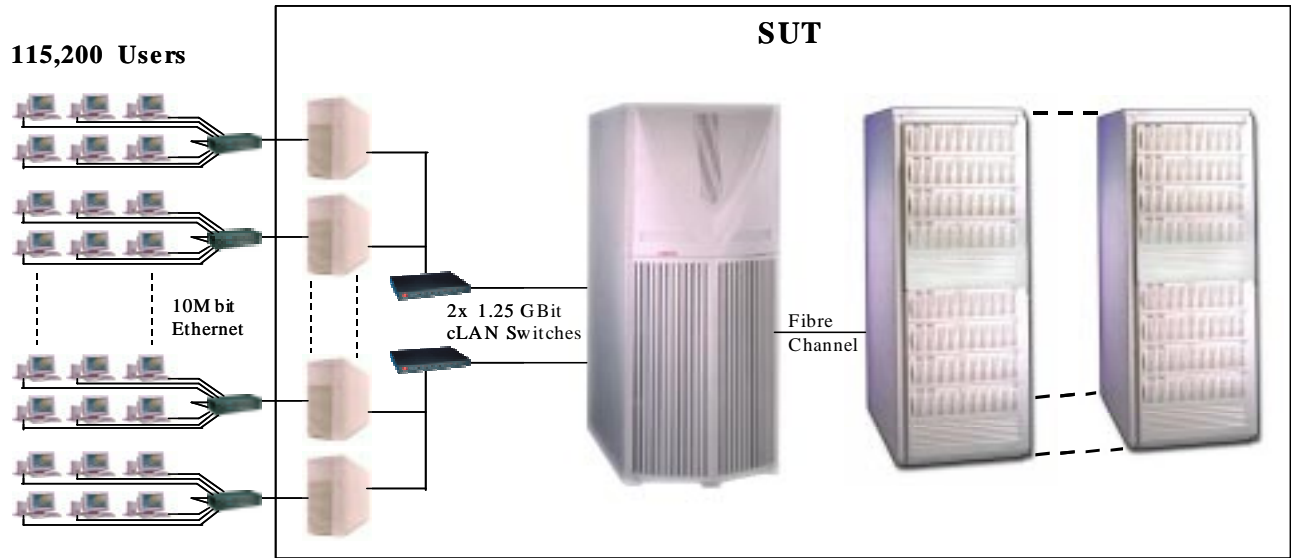
Executive Summary

The following pages contain the executive summary results of the benchmark.

Auditor

The benchmark configuration, environment, and methodology used to produce and validate the test results, along with the pricing model used to calculate the cost per tpmC, were audited by Lorna Livingtree of Performance Metrics, Inc. to verify compliance with the relevant TPC specification.

UNISYS		e-@ction Enterprise Server ES7000		TPC-C Rev. 5.0
				Report Date September 10, 2001
Total System Cost		TPC-C Throughput	Price Performance	Availability Date
\$3,363,483		141,138.44 tpmC	\$23.84 / tpmC	March 10, 2002
Processors	Database Manager	Operating System	Other Software	Number of Users
32 - Database 18 - Clients	Microsoft SQL Server 2000 Enterprise Edition	Microsoft Windows Datacenter Server Limited Edition	Windows 2000 Server w/ IIS 5.0 and COM+ Microsoft Visual C++	115,200



Clients (each of 9)
e-@ction ES2024
 2 x 1GHz Pentium® III
 w/ 256KB L2 cache,
 512MB memory,
 1 x 18GB internal disks,
 1 Inbuilt 10/100Mb
 Ethernet controller,
 1 x PCI cLAN controller

Enterprise Server
e-@ction ES7000
 32 x Pentium® III Xeon
 900MHz 2MB L2 cache,
 64 GB memory,
 1 PCI SCSI controller,
 2 x 18GB internal SCSI disk,
 9 x PCI Fibre Channel controller,
 2 x PCI cLAN controller

Storage
ESM7700
 919 x 18GB + 24 x 36GB
 Hot-Pluggable
 External Fibre Channel Disks
 48 x FC RAID Controllers

System Components	Enterprise Server		Clients (each of 9)	
	Quantity	Type	Quantity	Type
Processors	32	900MHz Pentium® III Xeon with 2MB Level 2 Cache	2	1GHz Pentium® III with 256KB Level 2 Cache
Memory	1	64 GB	1	512 MB
Disk Controllers	9	PCI Fibre Channel	1	Inbuilt SCSI
	1	PCI SCSI		
Disk Drives	919	18GB FC (17.0 GB useable)	1	18GB SCSI (16.9 GB useable)
	24	36GB FC (34.0 GB useable)		
	2	18GB SCSI (17.0 GB useable)		
Total Storage		15.6 TB		16.9 GB
Tape / CD-ROM	1	SCSI CD-ROM	1	IDE CD-ROM
Terminals	1	Console Monitor	1	Console Monitor

UNISYS	e-@ction Enterprise Server ES7000		TPC-C Rev. 5.0				
			Report Date September 10, 2001				
Description	Part No.	Third Party		Unit Price	Qty.	Extended Price	3 yr.Maint. Price
		Brand	Price				
Server Hardware:							
SYS: ES7000, 2 Domain, 32P, 8 GB, 2 MSUs, 2 DIBs	ES73251-GDN		1	\$521,333	1	\$521,333	
MEM: 16GB, 1 GB DIMMs	RAM7-16G		1	\$81,600	3	\$244,800	
MEM: 8GB, 1 GB DIMMs	RAM7-8GB		1	\$43,520	1	\$43,520	
BRD: Memory Carrier Board, 0 Mem.	MSU2000-MBD		1	\$10,880	2	\$21,760	
IO: Module, PCI Enclosure w/ BID Motherboard	PCI2053-12		1	\$2,992	6	\$17,952	
IO: Module, PCI Enclosure 3.3V, 4 Card	PCI3-MOD		1	\$363	10	\$3,630	
CTRL: Fibre Channel HBA, 2-Port, 64-bit PCI	PCI6221-SW		1	\$2,063	9	\$18,567	
CTRL: cLAN, PCI Host Adapter & Cable	PCI10001-CXG		1	\$586	2	\$1,172	
CTRL: 2Ch SCSI, U160, PCI	PCI7602-P64		1	\$442	1	\$442	
DISK: 18GB, 10K SCSI LVD, SCA	HDM18110-CX1		1	\$568	2	\$1,136	
I/F: Monitor, 15-inch Color, Kybrd, Mse & Cable	ES70001-UIF		1	\$272	1	\$272	
MAINT: 3-Yr. Cmprhnsv-Gold Svc Wrrnty Upgrd	WID7323-CGS		1	\$39,038	1		\$39,038
Server Software:							
SYS MGT: ES7000 Value Add Software	ESS700030-N		3	\$4,533	1	\$4,533	\$3,564
O/S: Microsoft Windows Datacenter Server Lmtd Edtn, 1-32P	LED320032-EN		3	\$90,667	1	\$90,667	\$71,388
O/S: Windows Datacenter Annual Update, 1-32P	WM320032-EN		1	\$26,293	3		\$78,879
APP: SQL Server 2000 Entrprs Edtn, per Proc. Lcnsng	810-00846		2	\$16,541	32	\$529,312	\$6,285
				Server Subtotal		\$1,499,096	\$199,154
Storage Hardware:							
DISK: 18GB Drive, 10K FC, SCA	ESM18201-F96		1	\$615	919	\$565,185	Spared
DISK: 18GB Drive, 10K FC, SCA 10% spares*	ESM18201-F96		1	\$615	92		\$56,580
DISK: 36GB Drive, 10K FC, SCA	ESM36202-F94		1	\$928	24	\$22,272	Spared
DISK: 36GB Drive, 10K FC, SCA 10% spares*	ESM36202-F94		1	\$928	3		\$2,784
DAE: Expansion Enclsr, w/ 0 Disk	ESM7200-DAE		3	\$4,087	72	\$294,264	\$62,208
DPE: 2 RAID Cntrl, 512MB Cache ea., 0 Disk	ESM7700-DPE		3	\$29,361	24	\$704,664	\$147,744
SYS MGT: Storage Agent, Monitor, Call Home & ATF	ESM7700-W2K		3	\$1,661	1	\$1,661	\$864
CBL: FC, DB9->DB9 Conn's, 10m	CBL135-10		1	\$193	18	\$3,474	
PWR: Distribution Strip, 220V	SFR9-PWR		1	\$295	24	\$7,080	
CBL: Power, U.S. (Domestic), L6-20P	USE1936-LC6		1	\$56	24	\$1,344	
CAB: 36U x 19" x 34" Cabinet, e-@ction	RM361934-OFE		1	\$810	12	\$9,720	
DOOR: 36U x 19", Rear	RM3619-RDR		1	\$277	12	\$3,324	
PNL: 36U x 34" Side Skins, L&R	RM3634-SDS		1	\$221	12	\$2,652	
				Storage Subtotal		\$1,615,640	\$270,180
Client Hardware:							
SYS: ES2024 Tower, w/ 0 Proc., 0MB Mem	ES202141-GZN		1	\$1,547	9	\$13,923	
PROC:1x1GHz Pentium III/256KB Cache	CPU31000-256		1	\$884	18	\$15,912	
ACC: Voltage Regulator	VRM3-83		1	\$29	9	\$261	
MEM: 256 MB SDRAM PC133ECC Memory	DIM13368-256		1	\$221	18	\$3,978	
DISK: 18GB SCSI 3.5 Internal	ES2024-18G		1	\$553	9	\$4,977	
CTRL: cLAN, PCI Host Adapter & Cable	PCI10001-CXG		1	\$586	9	\$5,274	
MAINT: 3 Yr. Cmprhnsv-Gold Svc Warranty Upgrade	WI2003-CGS		1	\$1,326	9		\$11,934
MONITOR:15-inch Color	EVG2100-P		1	\$221	9	\$1,989	
Client Software:							
O/S: Microsoft Windows 2000 Server	C11-00821		2	\$738	9	\$6,642	Inc. above
ACC: Microsoft Visual C++ Professional 6.0	048-00317		2	\$549	1	\$549	Inc. above
				Client Subtotal		\$53,505	\$11,934
Connectivity							
SWITCH: cLAN, 8-port, 1.25Gbit	SWT50082-CXG		3	\$5,157	2	\$10,314	\$2,160
				Connectivity Subtotal		\$10,314	\$2,160
				Total		\$3,178,555	\$483,428
Comark and Unisys Pre-Pay Discounts						(\$283,127)	(\$15,373)
Notes:				Three Year Cost of Ownership: \$3,363,483			
1. 3rd Party Brand and Pricing: 1 = Comark, 2 = Microsoft, 3 = Comark Product Price, Unisys Maintenance Price				tpmC Rating: 141,138.44			
2. HW & SW maintenance figured at 24 x 7 w/ 4 hr. max. response time for spares.				\$ / tpmC: \$23.84			
3. * = 10% spare disks added in place of onsite service.							
Benchmark results and test methodology audited by Lorna Livingtree of Performance Metrics, Inc.							
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 assumption 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 benchmarks 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.							

Numerical Quantities Summary

MQTh, Computed Maximum Qualified Throughput: 141,138.44 tpmC

Response Times, in seconds

Transaction	90th %ile	Average	Maximum
New-Order	1.03	0.68	6.00
Payment	0.95	0.60	5.81
Order-Status	0.97	0.62	6.18
Delivery (interactive portion)	0.58	0.32	1.33
Delivery (deferred portion)	0.52	0.37	1.66
Stock-Level	1.35	0.89	3.00
Menu	0.59	0.32	1.44

Response time delay added for emulated components, in seconds

RT Response time	0.1
Menu Response time	0.1

Transaction Mix, in percent of total transactions

New Order	44.81%
Payment	43.07%
Order-Status	4.03%
Delivery (interactive portion)	4.03%
Stock-Level	4.03%

Keying / Think Time Times, in seconds

Transaction	Minimum	Average	Maximum
New Order	18.00 / 0.00	18.00 / 12.03	18.05 / 120.30
Payment	3.00 / 0.00	3.00 / 12.03	3.02 / 120.31
Order-Status	2.00 / 0.00	2.00 / 10.06	2.01 / 100.71
Delivery (interactive portion)	2.00 / 0.00	2.00 / 5.06	2.03 / 50.60
Stock-Level	2.00 / 0.00	2.00 / 5.06	2.03 / 50.60

Test Duration

Ramp up time	54 minutes
Measurement interval (M)	120 minutes
Transactions (all types) completed during measurement interval	37,788,695
Ramp-down time	10 minutes

Checkpointing:

Number of checkpoints	4
Checkpoint interval	30 minutes

Table of Contents

Abstract	iv
Table of Contents	viii
Preface	xii
0. General Items	0-1
0.1. Order and Titles	0-1
0.2. Executive Summary Statement	0-1
0.3. Numerical Quantities Summary	0-1
0.4. Application Code Disclosure	0-1
0.5. Benchmark Sponsor	0-2
0.6. Parameter Settings	0-2
0.7. Configuration Diagrams	0-2
1. Clause 1: Logical Database Design	1-1
1.1. Table Definitions	1-1
1.2. Physical Organization of the Database	1-1
1.3. Insert and/or Delete Operations	1-1
1.4. Partitioning	1-1
1.5. Replication, Duplication or Additions	1-1
2. Clause 2: Transaction & Terminal Profiles	2-1
2.1. Random Number Generation	2-1
2.2. Input/Output Screen Layout	2-1
2.3. Priced Terminal Feature Verification	2-1
2.4. Presentation Managers or Intelligent Terminal	2-1
2.5. Transaction Statistics	2-1
2.6. Queuing Mechanism of Delivery	2-2
3. Clause 3: Transaction & System Properties	3-1
3.1. Transaction System Properties (ACID)	3-1
3.2. Atomicity	3-1
3.2.1. Completed Transaction	3-1
3.2.2. Aborted Transactions	3-1
3.3. Consistency	3-2
3.4. Isolation	3-2

3.5.	Durability	3-2
3.5.1.	Loss of Log Disk and Loss of Data Disk.....	3-2
3.5.2.	Instantaneous Interruption and Loss of Memory.....	3-3
4.	Clause 4: Scaling & Database Population.....	4-1
4.1.	Initial Cardinality of Tables	4-1
4.2.	Constant Values	4-1
4.3.	Database Layout.....	4-2
4.4.	DBMS: Data Model and DBMS Interface/Access Language	4-2
4.5.	DBMS Partitions/Replications	4-2
4.6.	DBMS Space Requirements.....	4-2
5.	Clause 5: Performance Metrics & Response Time	5-1
5.1.	Measured Throughput (tpmC).....	5-1
5.2.	Response Times	5-1
5.3.	Keying and Think Times.....	5-1
5.4.	Response Time Frequency Distribution Curves	5-2
5.5.	Response Time versus Throughput Performance Curve	5-4
5.6.	New Order Think Time Frequency Distribution Curve.....	5-5
5.7.	New-Order Throughput vs. Time.....	5-5
5.8.	Determination of “Steady State”	5-6
5.9.	Work Performed During Steady State.....	5-6
5.10.	Measurement Interval Duration.....	5-6
5.11.	Regulation of Transaction Mix.....	5-7
5.12.	Transaction Statistics	5-7
5.13.	Checkpoint Statistics.....	5-8
6.	Clause 6: SUT, Driver & Communications Definition.....	6-1
6.1.	Remote Terminal Emulator (RTE) Description	6-1
6.2.	Loss of Terminal Connections	6-1
6.3.	Emulated Components	6-1
6.4.	Functional Diagrams	6-1
6.5.	Network Configuration.....	6-1
6.6.	Network Bandwidth	6-2
6.7.	Operator Intervention	6-2
7.	Clause 7: Pricing	7-1
7.1.	Pricing.....	7-1
7.1.1.	System Pricing.....	7-1
7.1.2.	Maintenance Pricing.....	7-1
7.1.3.	Discounts.....	7-1

7.2.	Availability.....	7-2
7.3.	Measured tpmC, Pricing, Price/Performance, and Availability Date	7-2
7.4.	Country-Specific Pricing	7-2
7.5.	Usage Pricing	7-2
8.	Clause 8 : Full Disclosure Availability	8-1
8.1.	Availability.....	8-1
9.	Clause 9 : Audit.....	9-1
9.1.	Auditor's Report.....	9-1
Appendix A -	Client Source	A-1
Appendix B -	Database Design	B-1
Appendix C -	Tunable Parameters	C-1
Appendix D -	RTE Code	D-1
Appendix E -	Disk Storage	E-1
Appendix F -	Third-Party Price Quotations.....	F-1

Figures

Figure 0.1: Benchmarked Configuration.....	0-3
Figure 0.2: Priced Configuration.....	0-4
Figure 5.1: New Order Response Time Distribution	5-2
Figure 5.2: Payment Response Time Distribution	5-2
Figure 5.3: Order Status Response Time Distribution	5-3
Figure 5.4: Delivery Response Time Distribution	5-3
Figure 5.5: Stock Level Response Time Distribution	5-4
Figure 5.6: New Order Response Time versus Throughput.....	5-4
Figure 5.7: New Order Think Time Distribution	5-5
Figure 5.8: Throughput (tpmC) versus Time	5-5

Tables

Table 4.1: Initial Cardinality of Database Table	4-1
Table 4.2: Constant C for NURand.....	4-1
Table 4.3: Disk Cage Configuration.....	4-3
Table 4.4: Disk Management Configuration	4-5
Table 5.1: Response Time Data	5-1
Table 5.2: Keying Times	5-1
Table 5.3: Think Times	5-1
Table 5.4: Transaction Statistics	5-7
Table 5.5: Checkpoint Start Time and Duration	5-8

Document Structure

The TPC Benchmark C Standard Specification requires test sponsors to publish, submit to the TPC, and make available to the public, a full disclosure report for any result to be considered compliant with the specification. The required contents of the full disclosure report are specified in Clause 8.

This report is submitted to satisfy the specification's requirement for full disclosure. It documents the compliance of the benchmark implementation and execution reported for the Unisys e-@ction Enterprise Server ES7000 using Microsoft SQL Server 2000 Enterprise Edition on Microsoft Windows Datacenter Server Limited Edition.

TPC Benchmark C Overview

The TPC Benchmark™ C Standard Specification Revision 5.0 was developed by the Transaction Processing Performance Council (TPC). It is the intent of the TPC to develop a suite of benchmarks to measure the performance of computer systems executing a wide range of applications. Unisys and Microsoft Corporations are active participants in the TPC to define and develop such a suite of benchmarks.

TPC Benchmark™ C (TPC-C) is an 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 on 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 environments, 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 any other environment are not recommended.

0.1. Order and Titles

The order and titles of sections in the Test Sponsor's Full Disclosure report must correspond with the order and titles of sections from the TPC-C standard specification (i.e., this document). The intent is to make it as easy as possible for readers to compare and contrast material in different Full Disclosure reports. (8.1.1.1)

The order and titles of the sections in this report correspond with those from the TPC-C standard specification.

0.2. Executive Summary Statement

The TPC Executive Summary Statement must be included near the beginning of the Full Disclosure report. (8.1.1.2)

The TPC Executive Summary Statement is included near the beginning of this report.

0.3. Numerical Quantities Summary

The numerical quantities listed below must be summarized near the beginning of the Full Disclosure Report: (8.1.1.3)

- *measurement interval in minutes,*
- *number of checkpoints in the measurement interval,*
- *checkpoint interval in minutes,*
- *number of transactions (all types) completed within the measurement interval,*
- *computed Maximum Qualified Throughput in tpmC,*
- *• ninetieth percentile, average and maximum response times for the New-Order, Payment, Order-Status, Stock-Level, Delivery (deferred and interactive) and Menu transactions,*
- *time in seconds added to response time to compensate for delays associated with emulated components,*
- *percentage of transaction mix for each transaction type.*

These numerical quantities are summarized near the beginning of this report.

0.4. Application Code Disclosure

The applicable 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 and output functions. (8.1.1.4)

Appendix A contains the client application code used in this TPC-C benchmark. Appendix B contains the SQL stored procedures which implement the TPC-C transactions.

0.5. Benchmark Sponsor

A statement identifying the benchmark sponsor(s) and other participating companies must be provided. (8.1.1.5)

This TPC benchmark C was sponsored by Unisys Corporation. The benchmark test was developed by Microsoft and Unisys. The benchmark was conducted at Unisys, Mission Viejo, California.

0.6. 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 but not limited to: (8.1.1.6)

- *Data Base tuning options*
- *Recovery/commit options*
- *Consistency/locking options*
- *Operating system and application configuration parameters*

Appendix C contains the configuration and system parameters used in running these TPC-C tests. It also contains all the client and server OS and SQL Server tunable parameters.

0.7. Configuration Diagrams

Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences. This includes, but is not limited to: (8.1.1.7)

- *Number and type of processors.*
- *Size of allocated memory, and any specific mapping/partitioning of memory unique to the test.*
- *Number and type of disk units (and controllers, if applicable).*
- *Number of channels or bus connections to disk units, including their protocol type.*
- *Number of LAN (e.g., Ethernet) connections, including routers, workstations, terminals, etc., that were physically used in the test or are incorporated into the pricing structure (see Clause 8.1.8).*
- *Type and the run-time execution location of software components (e.g., DBMS, client processes, transaction monitors, software drivers, etc.).*

The Remote Terminal Emulator (RTE) software used for these TPC-C tests is proprietary to Unisys. The benchmarked configuration of the RTE and Unisys e-@ction Enterprise Server ES7000 is illustrated in Figure 0.1. Tables 4.3, 4.4 and 4.5 contain a detailed explanation of the disk configuration.

The priced configuration for the Unisys e-@ction Enterprise Server ES7000 is shown in Figure 0.2.

Figure 0.1: Benchmarked Configuration

Unisys e-@ction Enterprise Server ES7000 - Benchmarked Configuration

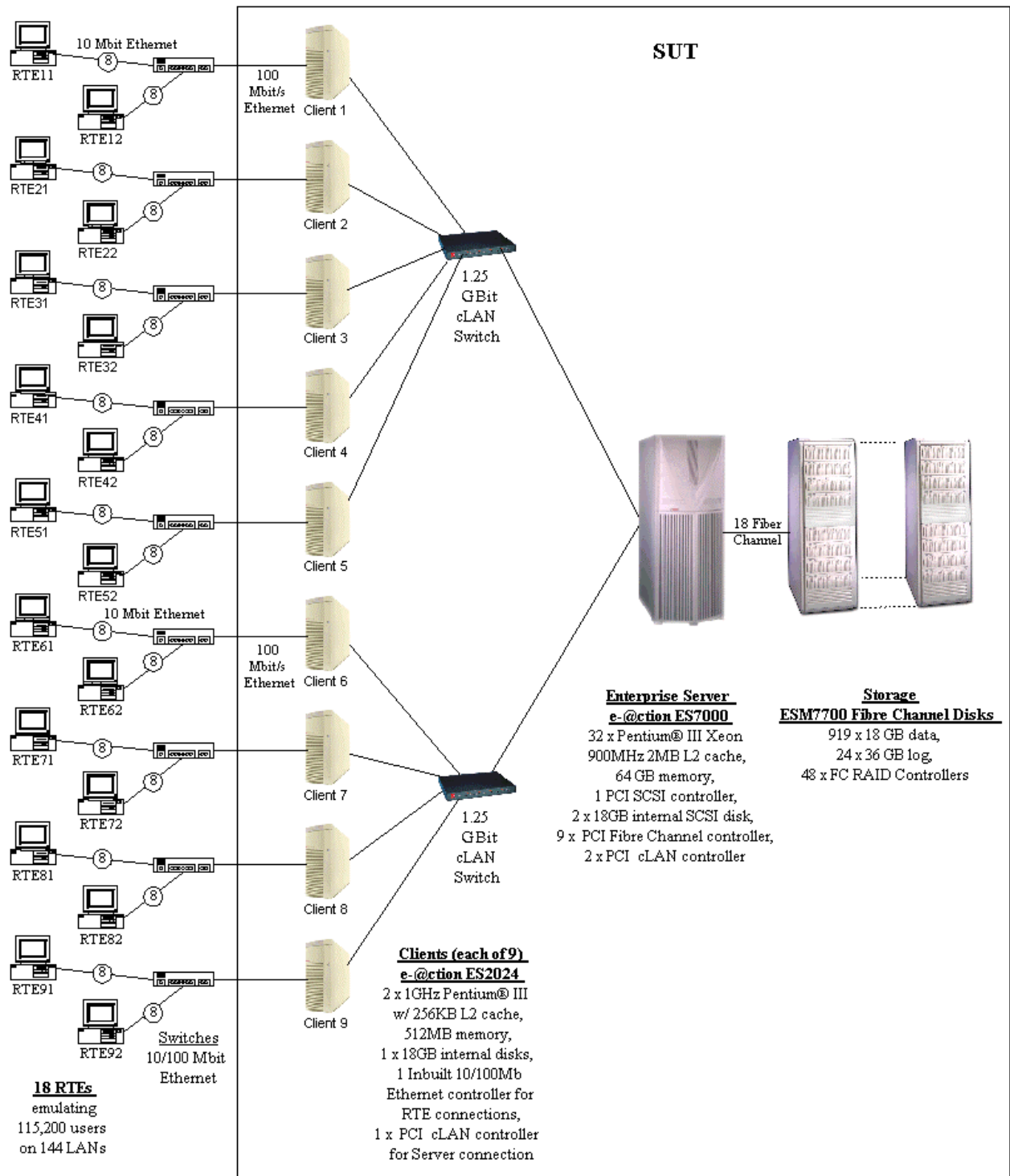
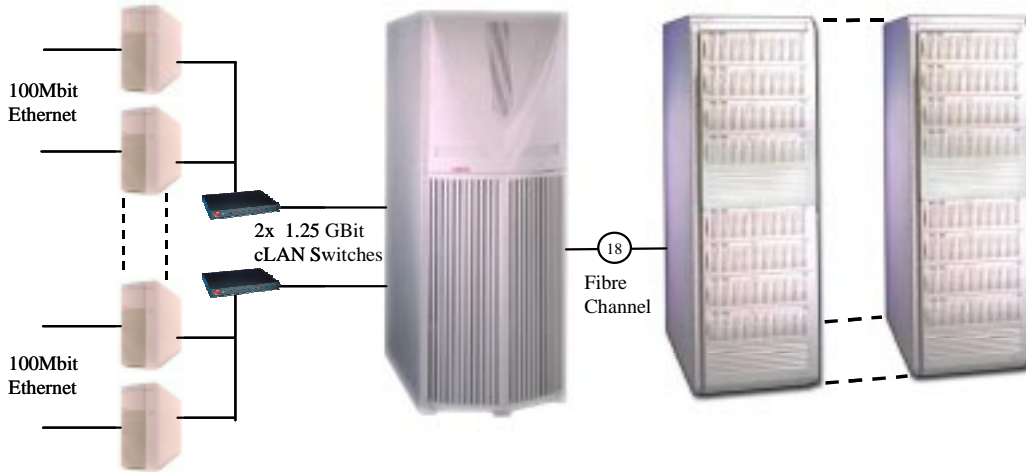


Figure 0.2: Priced Configuration

Unisys e-@ction Enterprise Server ES7000 - Priced Configuration



Clients (each of 9)
e-@ction ES2024
2 x 1GHz Pentium® III
w/ 256KB L2 cache,
512 MB memory,
1 x 18GB internal disks,
1 Inbuilt 10/100Mb
Ethernet controller,
1 x PCI cLAN controller

Enterprise Server
e-@ction ES7000
32 x Pentium® III Xeon
900MHz 2MB L2 cache,
64 GB memory,
1 PCI SCSI controller,
2 x 18GB internal SCSI disk,
9 x PCI Fibre Channel controller,
2 x PCI cLAN controller

Storage
ESM7700 Fibre Channel Disks
919 x 18 GB data,
24 x 36 GB log,
48 x FC RAID Controllers

1.

Clause 1: Logical Database Design

1.1. Table Definitions

Listings must be provided for all table definition statements and all other statements used to setup the database. (8.1.2.1)

Appendix B contains the SQL definitions of all the required database files, filegroups, tables, indexes and stored procedures, plus a listing of the program used to load the database and establish the required initial populations of each table.

1.2. Physical Organization of the Database

The physical organization of tables and indices, within the database, must be disclosed. (8.1.2.2)

The disk space was allocated to SQL Server according to the data in Tables 4.3, 4.4 and 4.5. The SQL definitions are contained in Appendix B.

1.3. Insert and/or 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 restriction 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 maximum key value for these new rows. (8.1.2.3)

There were no restrictions on insert and/or delete operations to any of the tables.

1.4. Partitioning

While there are few restrictions placed upon horizontal or vertical partitioning of tables and rows in the TPC-C benchmark (see Clause 1.6), any such partitioning must be disclosed. (8.1.2.4)

Partitioning was not used for any table in this implementation.

1.5. Replication, Duplication or Additions

Replication of tables, if used, must be disclosed (see Clause 1.4.6). (8.1.2.5)

Additional and/or duplicate attributes in any table must be disclosed along with a statement on the impact on performance (see Clause 1.4.7). (8.1.2.6)

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

2.

Clause 2: Transaction & Terminal Profiles

2.1. Random Number Generation

The method of verification for the random number generation must be disclosed. (8.1.3.1)

The drivers used the Unisys RTE program, which was independently audited. The initial population of the database was performed by the loader program from V4.21 of the Microsoft TPC-C toolkit, which was also independently audited. Furthermore, the auditor sampled various initial and runtime distributions produced by this implementation to verify correctness.

2.2. Input/Output Screen Layout

The actual layout of the terminal input/output screens must be disclosed. (8.1.3.2)

The screen layouts are based on those in Clauses 2.4.3, 2.5.3, 2.6.3, 2.7.3, and 2.8.3 of the TPC Benchmark C Standard Specification. There are some minor differences in appearance due to the use of a web client implementation.

2.3. 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). (8.1.3.3)

This was verified by the auditor.

2.4. Presentation Managers or Intelligent Terminal

Any usage of presentation managers or intelligent terminals must be explained. (8.1.3.4)

Application code running on the client implemented the TPC-C user interface. A listing of this code is included in Appendix A. No presentation manager was used on the client, as screen manipulation and data input/output was handled for each user by the Microsoft Internet Explorer web browser running on each user PC.

2.5. Transaction Statistics

The percentage of home and remote order-lines in the New-Order transactions must be disclosed. (8.1.3.5)

The percentage of New-Order transactions that were rolled back as a result of an unused item number must be disclosed. (8.1.3.6)

The number of items per order entered by New-Order transactions must be disclosed. (8.1.3.7)

The percentage of home and remote Payment transactions must be disclosed. (8.1.3.8)

The percentage of Payment and Order-Status transactions that used non-primary key (C_LAST) access to the database must be disclosed. (8.1.3.9)

The percentage of Delivery transactions that were skipped as a result of an insufficient number of rows in the NEW-ORDER table must be disclosed. (8.1.3.10)

The mix (i.e., percentages) of transaction types seen by the SUT must be disclosed. (8.1.3.11)

Table 5.4 in Section 5 contains all these statistics.

2.6. Queuing Mechanism of Delivery

The queuing mechanism used to defer execution of the Delivery transaction must be disclosed (8.1.3.12).

Deferred deliveries are queued by making an entry in an array within the application process (tpcc.dll) running on the client systems. Background threads within the application process asynchronously process the queued delivery transactions and log the results to a file upon completion.

3.

Clause 3: Transaction & System Properties

3.1. 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. (8.1.4.1)

The TPC Benchmark C Standard Specification defines a set of transaction processing system properties that a system under test (SUT) must support during the execution of the benchmark. Those properties are Atomicity, Consistency, Isolation, and Durability (ACID).

This section defines each of these properties, describes the steps taken to ensure that they were present during the test and describes a series of tests done to demonstrate compliance with the specification. All ACID property tests were executed successfully.

3.2. Atomicity

The system under test must guarantee that data base 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.

3.2.1. Completed Transaction

Perform the Payment transaction for a randomly selected warehouse, district, and customer (by customer number) and verify that the records in the CUSTOMER, DISTRICT, and WAREHOUSE tables have been changed appropriately.

The balances from a randomly selected warehouse, district, and customer row were retrieved by customer number from a script. A Payment transaction was submitted with the same warehouse, district and customer identifiers for a known amount. After completion of the Payment transaction, the balances of the selected warehouse, district, and customer were again retrieved to verify that the changes had been made correctly.

3.2.2. Aborted Transactions

Perform the Payment transaction for a randomly selected warehouse, district, and customer (by customer number) and substitute a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the records in the CUSTOMER, DISTRICT, and WAREHOUSE tables have NOT been changed.

The balances from a randomly selected warehouse, district, and customer row were retrieved by customer number from a script. A Payment transaction was submitted with the same warehouse, district and customer identifiers that issued a ROLLBACK command rather than a COMMIT. After the transaction completed, the balances of the selected warehouse, district, and customer were again retrieved to verify that no changes had been made to the database.

3.3. Consistency

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

The benchmark specification requires explicit demonstration of the following four consistency conditions:

1. The sum of the district balances in a warehouse is equal to the warehouse balance;
2. For each district, the next order id minus one is equal to maximum order id in the ORDER table and equal to the maximum new order id in the NEW ORDER table;
3. For each district, the maximum order id minus minimum order id in the ORDER table plus one equals the number of rows in the NEW-ORDER table for that district;
4. For each district, the sum of the order line counts in the ORDER table equals the number of rows in the ORDER-LINE table for that district;

In order to demonstrate this consistency, the following steps were taken:

1. Prior to the start of a benchmark run, the consistency of the database was verified by testing successfully conditions 1-4 described above with a script.
2. A run under full user load was executed for over 10 minutes with a checkpoint during the run.
3. After completion of that test, the consistency of the database was again verified by successfully testing using the same consistency script as in step 1.

3.4. 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.

The benchmark specification defines seven required tests to be performed to demonstrate that required levels of transaction isolation are met. These tests, described in Clauses 3.4.2.1 - 3.4.2.7, were all performed from a script and verified by the auditor. In Isolation Test 7, Case A was observed. In addition, the phantom tests and stock level tests were executed and verified to be successful.

3.5. Durability

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

Three durability tests were executed to satisfy the requirements of the specification. The tests for loss of memory and instantaneous interruption were combined and performed with a fully scaled database with all emulated users. The loss of log and loss of data tests were performed on the same database with 14,400 emulated users. To the best of our knowledge, these tests prove that the fully scaled configuration used for the throughput test would also meet all durability tests.

3.5.1. Loss of Log Disk and Loss of Data Disk

The following steps were taken with only 14,400 emulated users to demonstrate durability in the case of loss of a log disk and loss of a data disk:

1. The database was backed up to extra space on a backup volume.
2. The D_NEXT_O_ID fields for all rows in the district table were summed up to determine the initial count of orders present in the database.
3. The RTE was started with 14,400 users. On the driver systems, committed and rolled back New-Order transactions were recorded in a “success” file.
4. After five minutes of running at steady state, a hot-pluggable log disk was removed from the disk cabinet, with no effect on NT or SQL server.
5. After 5 additional minutes of operation, a hot-pluggable data disk was removed from the disk cabinet.
6. Windows 2000 and SQL Server encountered IO errors due to the missing disk and recorded these errors in the system event log and SQL Server error log, respectively. The RTEs also recorded errors.
7. First, the RTEs and clients were stopped, then SQL Server was used to backup the transaction log to the backup volume.
8. Next, scripts were executed to drop the database and all its files. Then, SQL Server was shutdown and the SUT shutdown.
9. A data disk was inserted in the disk cabinet to replace the one removed. The RAID controller was used to recreate the stripe set containing the new data disk. (The missing log drive was not replaced.)
10. The SUT was restarted, SQL Server was then restarted, and a new (empty) database created as part of the restore database process. That process loaded the initial database into the new database, but did not perform any recovery. Next the transaction log was restored, followed by transaction recovery. The latter step restored all committed transactions to the database.
11. Consistency condition 3 of Clause 3.3.2.3 was executed to verify database consistency.
12. Step 2 was repeated to determine the total number of orders. This number was subtracted from the count obtained previously in Step 2 to determine the number of additional orders added to the database.
13. The contents of the “success” files on the drivers were sampled to verify that the records in the “success” file for committed New-Order transactions had corresponding records in the ORDER table. Moreover, the counts were matched with those obtained in step 12.

3.5.2. Instantaneous Interruption and Loss of Memory

Instantaneous interruption and loss of memory tests were combined because the loss of power erased the contents of memory. This failure was induced by removing the primary power to the System Under Test while the benchmark was executing.

1. The D_NEXT_O_ID fields for all rows in the district table were summed up to determine the initial count of orders present in the database (count1).
2. On the driver systems, committed and rolled back New-Order transactions were recorded in a “success” file.
3. The benchmark was executed at full load with all emulated users for a minimum of 10 minutes.
4. The system’s primary power was then turned off.
5. After transaction failures were noted by the RTEs, the RTEs and clients were shutdown.
6. Power was restored to the SUT, the system rebooted, SQL Server was restarted, and automatic database recovery was performed. The database recovery used the transaction log to reapply all committed transactions and rollback any (in progress) uncommitted transactions, so that the database disks were correct.
7. After recovery finished, Consistency Condition 3 of Clause 3.3.2.3 (no gaps in NO_O_ID) was executed to verify that the database was consistent.
8. Finally, step 1 was repeated to determine the total number of orders (count2). Count2 minus count1 was not less than the number of committed New-Order records in the “success” file from the drivers.

4.

Clause 4: Scaling & Database Population

4.1. Initial Cardinality of Tables

The Cardinality (e.g., the number of rows) of each table, as it existed at the start of the benchmark run (see Clause 4.2), must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted (see Clause 4.2.2), the cardinality of the WAREHOUSE table as initially configured and the number of rows deleted must be disclosed. (8.1.5.1)

The TPC-C database for this test was configured with 11,808 warehouses. The cardinality of each table in the database is listed in Table 4.1

Table 4.1: Initial Cardinality of Database Table

Table	Occurrences
Warehouse	11,808
District	118,080
Customer	354,240,000
History	354,240,000
Order	354,240,000
New-Order	106,272,000
Order Line	3,542,392,472
Stock	1,180,800,000
Item	100,000

288 upper warehouses were deleted and not used while executing the measurement runs.

4.2. Constant Values

The following values were used as the constant C input values to the NURand function during Build and Run time for this implementation.

Table 4.2: Constant C for NURand

Function	Value
C_LAST (Build)	123
C_LAST (Run)	208

4.3. Database Layout

The distribution of tables and logs across all media must be explicitly depicted for the tested and priced systems. (8.1.5.2)

Tables 4.3 and 4.4 list the distribution of the database over 919 18GB disks and the transaction log over 12 mirrored pairs of 36GB disks for the benchmark configuration. In addition, there were two 18GB disks in the host server, one containing Windows Datacenter Server Limited Edition and SQL Server 2000 code and the Master database plus the paging file. Unused-space partitions on the database drives were used for database backup. These partitions were not used during the measurements. Otherwise, the tested and priced disk configurations were identical.

4.4. DBMS: Data Model and DBMS Interface/Access Language

A statement must be provided that describes: (8.1.5.3)

1. *The data model implemented by the DBMS used (e.g., relational, network, hierarchical).*
2. *The database interface (e.g., embedded, call level) and access language (e.g., SQL, DL/1, COBOL, read/write) used to implement the TPC-C transactions. 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 2000 is a relational DBMS.

The client software interfaced to SQL Server through Stored Procedures invoked through ODBC calls embedded in the C application code.

4.5. DBMS Partitions/Replications

The mapping of database partitions/replications must be explicitly described. (8.1.5.4)

No table partitioning or replication was done.

4.6. DBMS Space Requirements

Details of the 60 day space computation along with proof that the database is configured to sustain 8 hours of growth for dynamic tables (Order, Order-line, and History) must be disclosed (see Clause 4.2.3) (8.1.5.5).

Appendix E lists the space requirements for the 60-day space as well as the logical log space for eight hours.

Table 4.3: Disk Cage Configuration (Continued)

Disk Cage Configuration																	
HBA	Port	ESM7700 DPE	SP	Cage #	RAID Level	Lun Capacity	Id 0	Id 1	Id 2	Id 3	Id 4	Id 5	Id 6	Id 7	Id 8	Id 9	
6	A	15	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
	B	16	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
		17	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
7	A	18	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
	B	19	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
		20	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
8	A	21	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
	B	22	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
		23	A	1	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				2	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB		
			B	3	0	165.39GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	
				4	0	148.85GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	18GB	Empty	
9	A	24	A	1	10	197.61GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	
				2	10	197.61GB	36GB	36GB	Empty	Empty	Empty	Empty	Empty	Empty	Empty		
			B	3	10	197.61GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	
				4	10	197.61GB	36GB	36GB	Empty	Empty	Empty	Empty	Empty	Empty	Empty	Empty	

Table 4.4: Disk Management Configuration

Disk Management Configuration								
HBA	Port	ESM7700 DPE	Logical Disk#	Partition 1 Size=7306MB FS=Raw Junction Point	Partition 2 Size=3505MB FS=Raw Junction Point	Partition 3 Size=8MB FS=NTFS Label	Partition 4 Size=155GB FS=NTFS File Name	
1	A	1	2	C:\BigDb\cabM\cs4	C:\BigDb\cabM\misc4	M4	Unallocated	
			3	C:\BigDb\cabI\cs1	C:\BigDb\cabI\misc1	I1	Unallocated	
			4	C:\BigDb\cabI\cs2	C:\BigDb\cabI\misc2	I2	Unallocated	
			5	C:\BigDb\cabI\cs3	C:\BigDb\cabI\misc3	I3	Unallocated	
			6	C:\BigDb\cabI\cs4	C:\BigDb\cabI\misc4	I4	C:\BigDb\dbBackups\backup8i	
	B	2	7	C:\BigDb\cabM\cs1	C:\BigDb\cabM\misc1	M1	Unallocated	
			8	C:\BigDb\cabM\cs2	C:\BigDb\cabM\misc2	M2	Unallocated	
			9	C:\BigDb\cabM\cs3	C:\BigDb\cabM\misc3	M3	Unallocated	
2	A	3	10	C:\BigDb\cabJ\cs1	C:\BigDb\cabJ\misc1	J1	Unallocated	
			11	C:\BigDb\cabJ\cs2	C:\BigDb\cabJ\misc2	J2	Unallocated	
			12	C:\BigDb\cabJ\cs3	C:\BigDb\cabJ\misc3	J3	Unallocated	
			13	C:\BigDb\cabJ\cs4	C:\BigDb\cabJ\misc4	J4	Unallocated	
	B	4		14	C:\BigDb\cabB\cs1	C:\BigDb\cabB\misc1	B1	Unallocated
				15	C:\BigDb\cabB\cs2	C:\BigDb\cabB\misc2	B2	Unallocated
				16	C:\BigDb\cabB\cs3	C:\BigDb\cabB\misc3	B3	Unallocated
				17	C:\BigDb\cabB\cs4	C:\BigDb\cabB\misc4	B4	Unallocated
		5		18	C:\BigDb\cabB\cs5	C:\BigDb\cabB\misc5	B5	Unallocated
				19	C:\BigDb\cabB\cs6	C:\BigDb\cabB\misc6	B6	Unallocated
				20	C:\BigDb\cabB\cs7	C:\BigDb\cabB\misc7	B7	Unallocated
				21	C:\BigDb\cabB\cs8	C:\BigDb\cabB\misc8	B8	C:\BigDb\dbBackups\backup8b
3	A	6	22	C:\BigDb\cabN\cs1	C:\BigDb\cabN\misc1	N1	Unallocated	
			23	C:\BigDb\cabN\cs2	C:\BigDb\cabN\misc2	N2	Unallocated	
			24	C:\BigDb\cabN\cs3	C:\BigDb\cabN\misc3	N3	Unallocated	
			25	C:\BigDb\cabN\cs4	C:\BigDb\cabN\misc4	N4	Unallocated	
	B	7		26	C:\BigDb\cabC\cs1	C:\BigDb\cabC\misc1	C1	Unallocated
				27	C:\BigDb\cabC\cs2	C:\BigDb\cabC\misc2	C2	Unallocated
				28	C:\BigDb\cabC\cs3	C:\BigDb\cabC\misc3	C3	Unallocated
				29	C:\BigDb\cabC\cs4	C:\BigDb\cabC\misc4	C4	Unallocated
		8		30	C:\BigDb\cabC\cs5	C:\BigDb\cabC\misc5	C5	Unallocated
				31	C:\BigDb\cabC\cs6	C:\BigDb\cabC\misc6	C6	Unallocated
				32	C:\BigDb\cabC\cs7	C:\BigDb\cabC\misc7	C7	Unallocated
				33	C:\BigDb\cabC\cs8	C:\BigDb\cabC\misc8	C8	C:\BigDb\dbBackups\backup8c
4	A	9	34	C:\BigDb\cabK\cs1	C:\BigDb\cabK\misc1	K1	Unallocated	
			35	C:\BigDb\cabK\cs2	C:\BigDb\cabK\misc2	K2	Unallocated	
			36	C:\BigDb\cabK\cs3	C:\BigDb\cabK\misc3	K3	Unallocated	
			37	C:\BigDb\cabK\cs4	C:\BigDb\cabK\misc4	K4	Unallocated	
	B	10		38	C:\BigDb\cabD\cs1	C:\BigDb\cabD\misc1	D1	Unallocated
				39	C:\BigDb\cabD\cs2	C:\BigDb\cabD\misc2	D2	Unallocated
				40	C:\BigDb\cabD\cs3	C:\BigDb\cabD\misc3	D3	Unallocated
				41	C:\BigDb\cabD\cs4	C:\BigDb\cabD\misc4	D4	Unallocated
		11		42	C:\BigDb\cabD\cs5	C:\BigDb\cabD\misc5	D5	Unallocated
				43	C:\BigDb\cabD\cs6	C:\BigDb\cabD\misc6	D6	Unallocated
				44	C:\BigDb\cabD\cs7	C:\BigDb\cabD\misc7	D7	Unallocated
				45	C:\BigDb\cabD\cs8	C:\BigDb\cabD\misc8	D8	C:\BigDb\dbBackups\backup8d
5	A	12	46	C:\BigDb\cabE\cs1	C:\BigDb\cabE\misc1	E1	Unallocated	
			47	C:\BigDb\cabE\cs2	C:\BigDb\cabE\misc2	E2	Unallocated	
			48	C:\BigDb\cabE\cs3	C:\BigDb\cabE\misc3	E3	Unallocated	
			49	C:\BigDb\cabE\cs4	C:\BigDb\cabE\misc4	E4	Unallocated	
		13		50	C:\BigDb\cabE\cs5	C:\BigDb\cabE\misc5	E5	Unallocated
				51	C:\BigDb\cabE\cs6	C:\BigDb\cabE\misc6	E6	Unallocated
				52	C:\BigDb\cabE\cs7	C:\BigDb\cabE\misc7	E7	Unallocated
				53	C:\BigDb\cabE\cs8	C:\BigDb\cabE\misc8	E8	C:\BigDb\dbBackups\backup8e
	B	14	54	C:\BigDb\cabO\cs1	C:\BigDb\cabO\misc1	O1	Unallocated	
			55	C:\BigDb\cabO\cs2	C:\BigDb\cabO\misc2	O2	Unallocated	
			56	C:\BigDb\cabO\cs3	C:\BigDb\cabO\misc3	O3	Unallocated	
			57	C:\BigDb\cabO\cs4	C:\BigDb\cabO\misc4	O4	Unallocated	

Table 4.4: Disk Management Configuration (Continued)

Disk Management Configuration							
HBA	Port	ESM7700 DPE	Logical Disk#	Partition 1 Size=7306MB FS=Raw Junction Point	Partition 2 Size=3505MB FS=Raw Junction Point	Partition 3 Size=8MB FS=NTFS Label	Partition 4 Size=155GB FS=NTFS File Name
6	A	15	58	C:\BigDb\cabL\cs1	C:\BigDb\cabL\misc1	L1	Unallocated
			59	C:\BigDb\cabL\cs2	C:\BigDb\cabL\misc2	L2	Unallocated
			60	C:\BigDb\cabL\cs3	C:\BigDb\cabL\misc3	L3	Unallocated
			61	C:\BigDb\cabL\cs4	C:\BigDb\cabL\misc4	L4	Unallocated
	B	16	62	C:\BigDb\cabF\cs1	C:\BigDb\cabF\misc1	F1	Unallocated
			63	C:\BigDb\cabF\cs2	C:\BigDb\cabF\misc2	F2	Unallocated
			64	C:\BigDb\cabF\cs3	C:\BigDb\cabF\misc3	F3	Unallocated
			65	C:\BigDb\cabF\cs4	C:\BigDb\cabF\misc4	F4	Unallocated
		17	66	C:\BigDb\cabF\cs5	C:\BigDb\cabF\misc5	F5	Unallocated
			67	C:\BigDb\cabF\cs6	C:\BigDb\cabF\misc6	F6	Unallocated
			68	C:\BigDb\cabF\cs7	C:\BigDb\cabF\misc7	F7	Unallocated
			69	C:\BigDb\cabF\cs8	C:\BigDb\cabF\misc8	F8	C:\BigDb\dbBackups\backup8f
7	A	18	70	C:\BigDb\cabP\cs1	C:\BigDb\cabP\misc1	P1	Unallocated
			71	C:\BigDb\cabP\cs2	C:\BigDb\cabP\misc2	P2	Unallocated
			72	C:\BigDb\cabP\cs3	C:\BigDb\cabP\misc3	P3	Unallocated
			73	C:\BigDb\cabP\cs4	C:\BigDb\cabP\misc4	P4	Unallocated
	B	19	74	C:\BigDb\cabG\cs1	C:\BigDb\cabG\misc1	G1	Unallocated
			75	C:\BigDb\cabG\cs2	C:\BigDb\cabG\misc2	G2	Unallocated
			76	C:\BigDb\cabG\cs3	C:\BigDb\cabG\misc3	G3	Unallocated
			77	C:\BigDb\cabG\cs4	C:\BigDb\cabG\misc4	G4	Unallocated
		20	78	C:\BigDb\cabG\cs5	C:\BigDb\cabG\misc5	G5	Unallocated
			79	C:\BigDb\cabG\cs6	C:\BigDb\cabG\misc6	G6	Unallocated
			80	C:\BigDb\cabG\cs7	C:\BigDb\cabG\misc7	G7	Unallocated
			81	C:\BigDb\cabG\cs8	C:\BigDb\cabG\misc8	G8	C:\BigDb\dbBackups\backup8g
8	A	21	82	C:\BigDb\cabA\cs1	C:\BigDb\cabA\misc1	A1	Unallocated
			83	C:\BigDb\cabA\cs2	C:\BigDb\cabA\misc2	A2	Unallocated
			84	C:\BigDb\cabA\cs3	C:\BigDb\cabA\misc3	A3	Unallocated
			85	C:\BigDb\cabA\cs4	C:\BigDb\cabA\misc4	A4	Unallocated
	B	22	86	C:\BigDb\cabH\cs1	C:\BigDb\cabH\misc1	H1	Unallocated
			87	C:\BigDb\cabH\cs2	C:\BigDb\cabH\misc2	H2	Unallocated
			88	C:\BigDb\cabH\cs3	C:\BigDb\cabH\misc3	H3	Unallocated
			89	C:\BigDb\cabH\cs4	C:\BigDb\cabH\misc4	H4	Unallocated
		23	90	C:\BigDb\cabH\cs5	C:\BigDb\cabH\misc5	H5	Unallocated
			91	C:\BigDb\cabH\cs6	C:\BigDb\cabH\misc6	H6	Unallocated
			92	C:\BigDb\cabH\cs7	C:\BigDb\cabH\misc7	H7	Unallocated
			93	C:\BigDb\cabH\cs8	C:\BigDb\cabH\misc8	H8	C:\BigDb\dbBackups\backup8h
HBA	Port	ESM7700 DPE	Logical Disk#	Partition 1 Size=98GB FS=Raw Junction Point	Partition 2 Size=8MB Type=NTFS Label	Partition 3 Size=98GB	
9	A	24	94	C:\BigDb\dbLogs\tpcclog1	A1A2	Unallocated	
			95	C:\BigDb\dbLogs\tpcclog2	A3A4	Unallocated	
System Disks			Logical Disk#	Partition 1 Drive Letter	File System		
17 GB SCSI disk			0	C: System	NTFS		
17 GB SCSI disk			1	D: No label	NTFS		
CD-ROM			CD-ROM 0	F:			

5. Clause 5: Performance Metrics & Response Time

5.1. Measured Throughput (tpmC)

Measured tpmC must be reported. (8.1.6.1)

The measured tpmC was 141,138.44.

5.2. Response Times

Ninetieth percentile, maximum and average response times must be reported for all transaction types as well as for the Menu response time. (8.1.6.2)

Table 5.1: Response Time Data

Transaction	Average	Maximum	90th %ile
New-Order	0.68	6.00	1.03
Payment	0.60	5.81	0.95
Order Status	0.62	6.18	0.97
Delivery (interactive)	0.32	1.33	0.58
Delivery (deferred)	0.37	1.66	0.52
Stock-Level	0.89	3.00	1.35
Menu	0.32	1.44	0.59

5.3. Keying and Think Times

The minimum, the average, and the maximum keying and think times must be reported for each transaction type. (8.1.6.3)

Table 5.2: Keying Times

Transaction	Minimum	Average	Maximum
New-Order	18.00	18.00	18.05
Payment	3.00	3.00	3.02
Order Status	2.00	2.00	2.01
Delivery (interactive)	2.00	2.00	2.03
Stock-Level	2.00	2.00	2.03

Table 5.3: Think Times

Transaction	Minimum	Average	Maximum
New-Order	0.00	12.03	120.30
Payment	0.00	12.03	120.31
Order Status	0.00	10.06	100.71
Delivery (interactive)	0.00	5.06	50.60
Stock-Level	0.00	5.06	50.60

5.4. Response Time Frequency Distribution Curves

Response Time frequency distribution curves (see Clause 5.6.1) must be reported for each transaction type. (8.1.6.4)

Figure 5.1: New Order Response Time Distribution

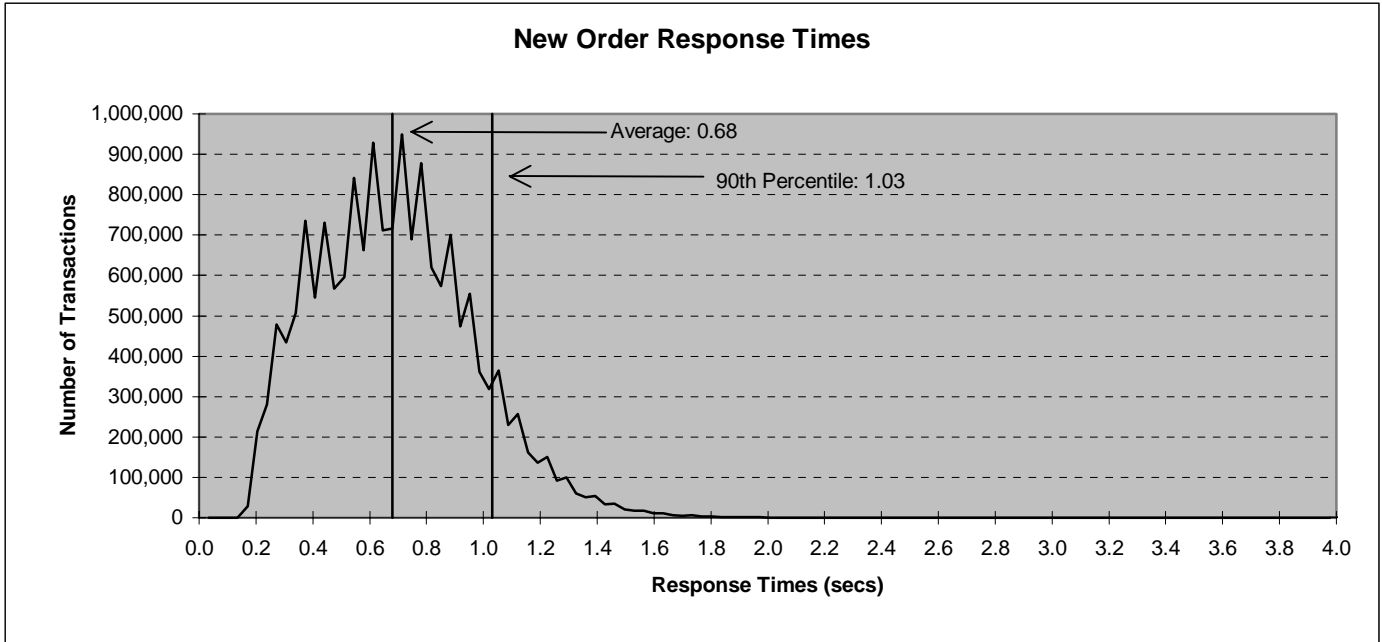


Figure 5.2: Payment Response Time Distribution

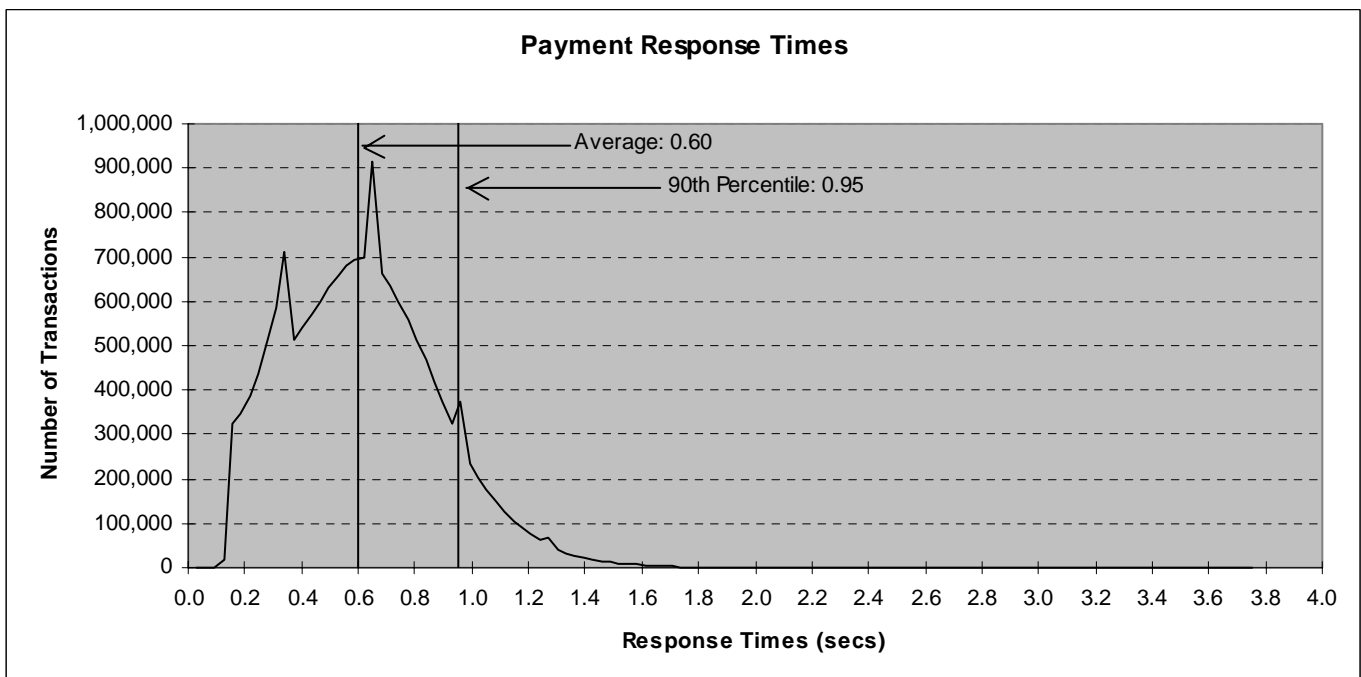


Figure 5.3: Order Status Response Time Distribution

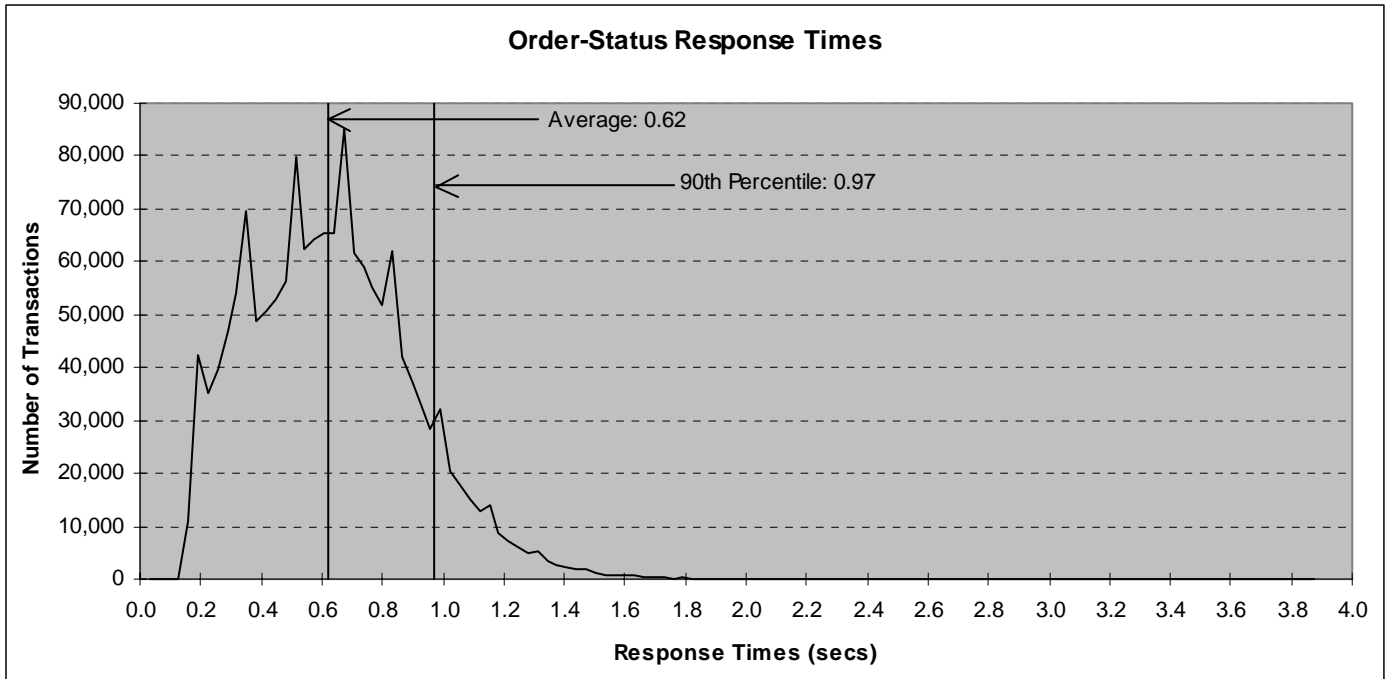


Figure 5.4: Delivery Response Time Distribution

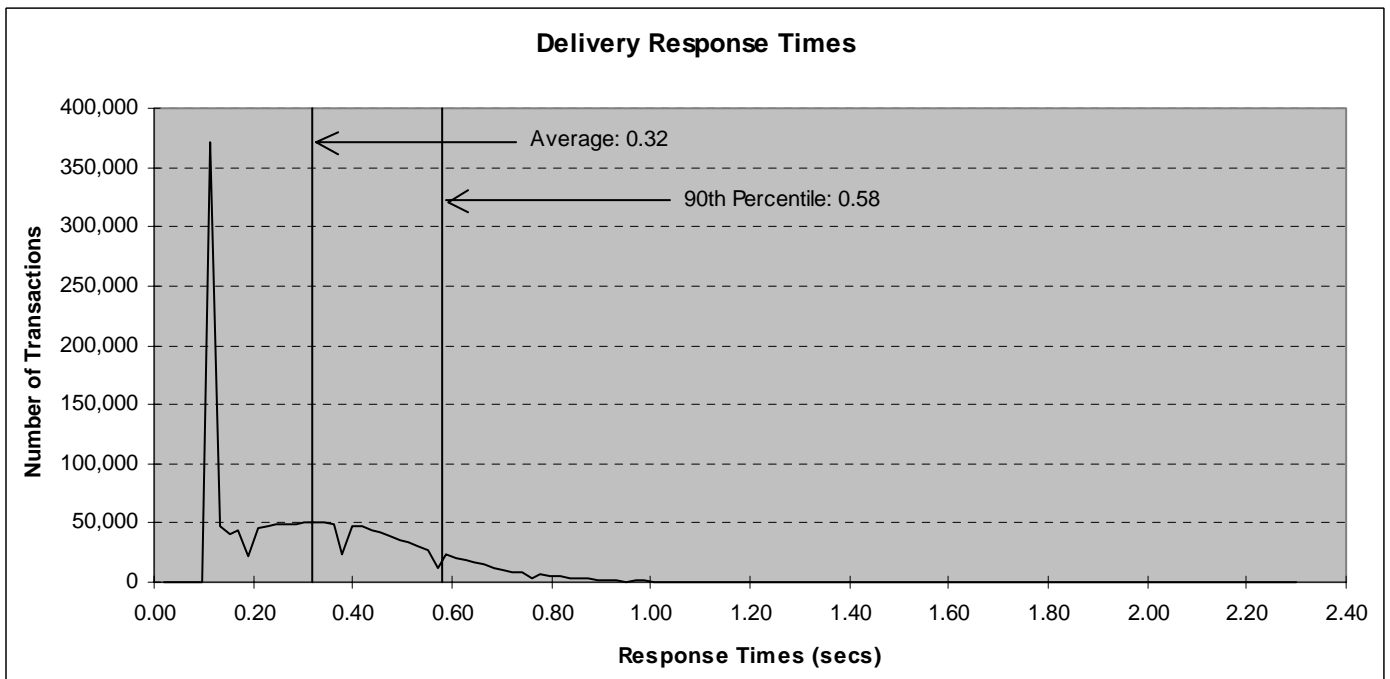
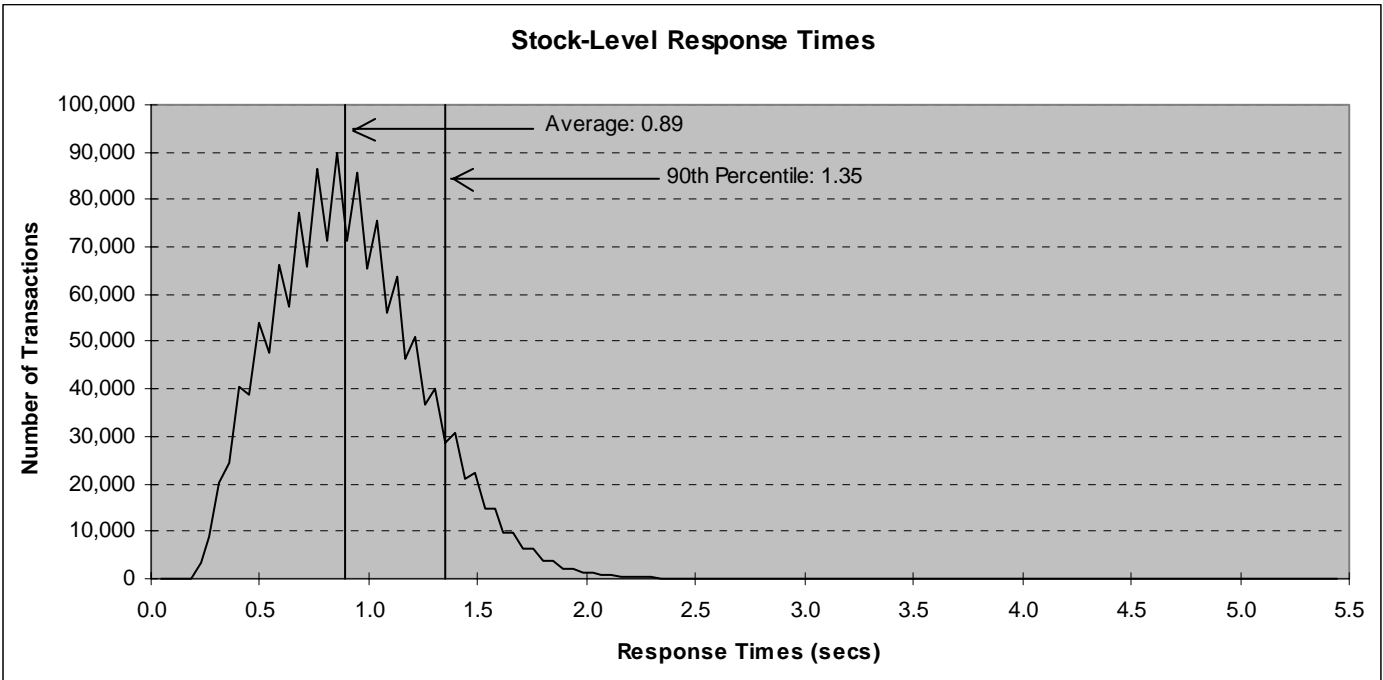


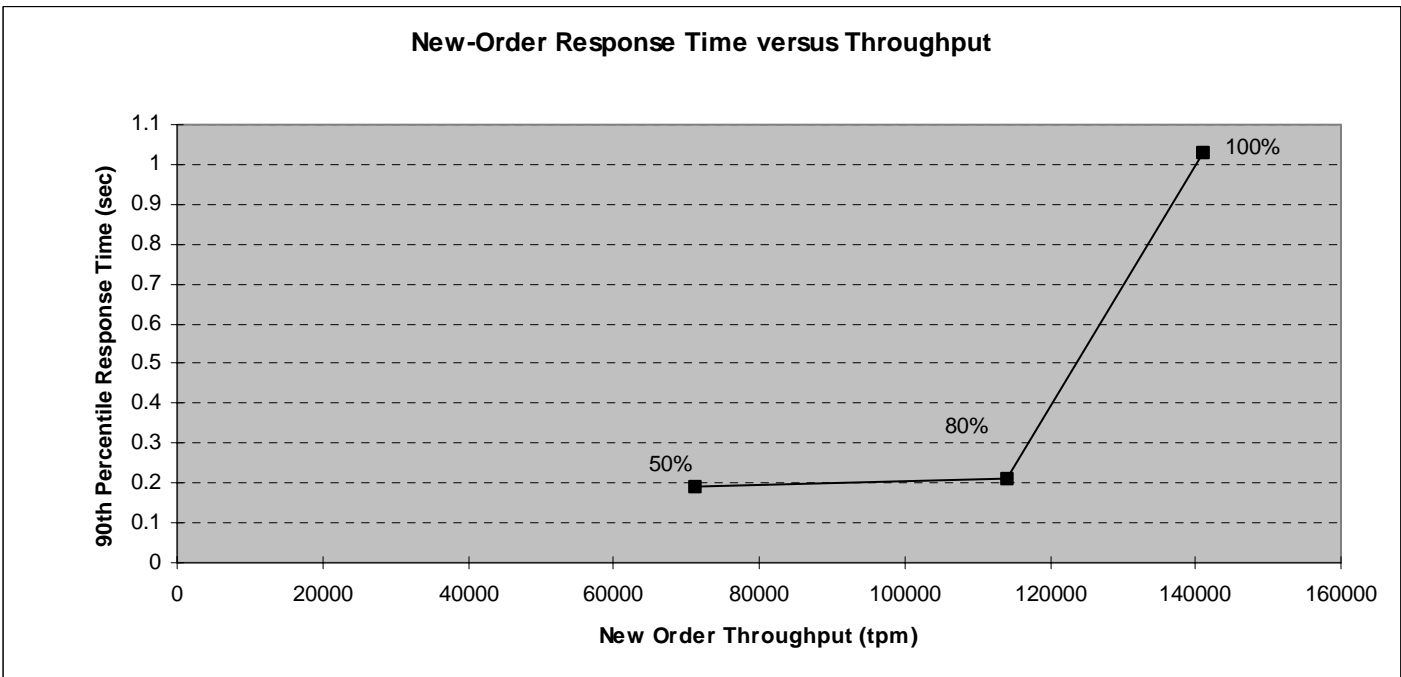
Figure 5.5: Stock Level Response Time Distribution



5.5. Response Time versus Throughput Performance Curve

The performance curve for response times versus throughput (Clause 5.6.2) must be reported for the New-Order transaction. (8.1.6.5)

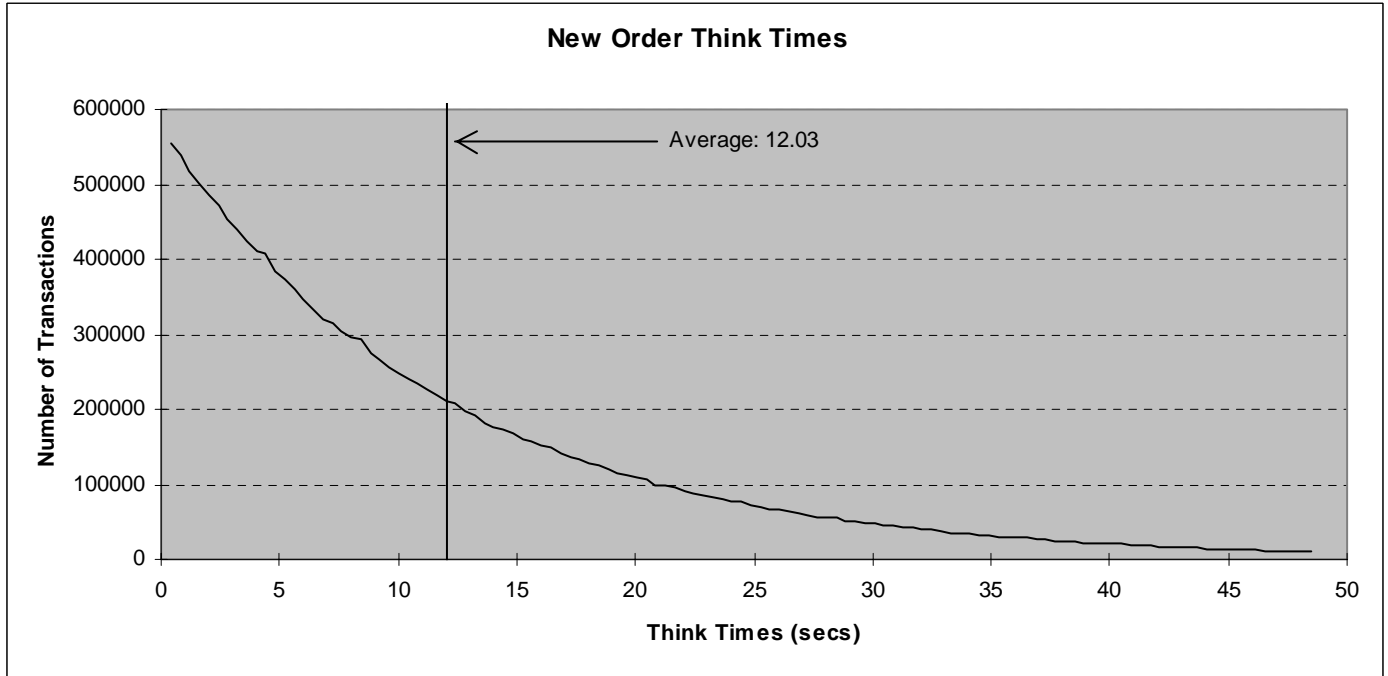
Figure 5.6: New Order Response Time versus Throughput



5.6. New Order Think Time Frequency Distribution Curve

Think Time frequency distribution curve (see Clause 5.6.3) must be reported for the New-Order transaction. (8.1.6.6)

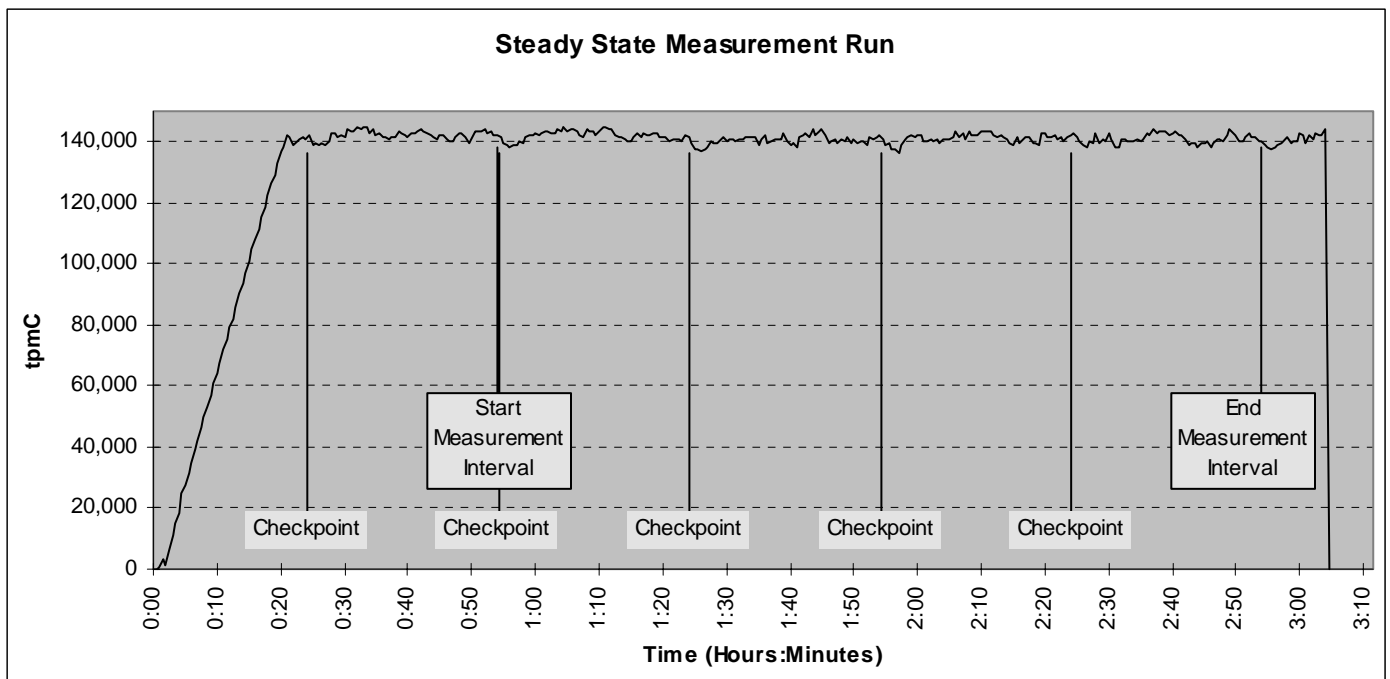
Figure 5.7: New Order Think Time Distribution



5.7. New-Order Throughput vs. Time

A graph of throughput versus elapsed time (Clause 5.6.4) must be reported for the New-Order transaction. (8.1.6.8)

Figure 5.8: Throughput (tpmC) versus Time



5.8. Determination of “Steady State”

The method used to determine that the SUT had reached a steady state prior to commencing the measurement interval (see Clause 5.5) must be described. (8.1.6.9)

The transaction throughput rate (tpmC) and response time were relatively constant after the initial ‘ramp up’ period. The throughput and response time behavior were determined by examining data reported for each 30-second interval over the duration of the benchmark. Ramp-up and steady state are discernible in the graph presented in Figure 5.8.

5.9. 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. (8.1.6.10)

The RTE selects a transaction type from the menu and prepares to request the appropriate blank form. A timestamp is taken before the form request is sent and after the response is returned. The difference between the two is saved off as the menu response time. The RTE then generates input data for the transaction to create a completed form and waits the appropriate key time. A timestamp is taken before the completed form is sent and after the response is returned. The difference between these two is saved off as the transaction response time. Both response times are padded with a 0.1 second delay per spec to account for the web browser delay. The appropriate transaction data and response times are logged and the RTE waits the required think time interval before repeating the process. Each RTE driver maintains its own log file. Log file contents are consolidated for the reports.

The RTE emulates web browsers (not terminals) in this client-server implementation. The RTE sends and receives HTML formatted data using HTTP through Ethernet LANs to a client application running on the client machine. The client application processes the request, sends the transaction to a COM+ component, waits for the transaction response, and returns an appropriately formatted HTML form back to the (emulated) web browser (RTE). When activated, the COM+ component calls a stored procedure on the data base server using Microsoft SQL Server ODBC, collects the response, and returns the result to the requestor.

To perform checkpoints at specific intervals, SQL Server’s recovery interval was set to four times the desired checkpoint duration, and a utility was written to schedule checkpoints at parameter-specified intervals and record the start and end time of each checkpoint. The checkpoint script was started manually on one of the client machines after the RTE had all users logged in and sending transactions and a steady state had been achieved.

At each checkpoint, SQL Server wrote to disk all database pages in memory that had been updated but not yet physically written to the disk. Upon completion of the checkpoint, SQL Server also wrote records to the error log indicating that a checkpoint had completed.

5.10. Measurement Interval Duration

A statement of the duration of the measurement interval for the reported Maximum Qualified Throughput (tpmC) must be included. (8.1.6.12)

The measurement interval was 120 minutes.

5.11. 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. (8.1.6.13)

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

5.12. Transaction Statistics

The percentage of the total mix for each transaction type must be disclosed. (8.1.6.14)

The percentage of New-Order transactions rolled back as a result of invalid item number must be disclosed. (8.1.6.15)

The average number of order-lines entered per New-Order transaction must be disclosed. (8.1.6.16)

The percentage of remote order-lines entered per New-Order transaction must be disclosed. (8.1.6.17)

The percentage of remote Payment transactions must be disclosed. (8.1.6.18)

The percentage of customer selections by customer last name in the Payment and Order-Status transactions must be disclosed. (8.1.6.19)

The percentage of Delivery transactions skipped due to there being fewer than necessary orders in the New-Order table must be disclosed. (8.1.6.20)

Table 5.4 shows this information.

Table 5.4: Transaction Statistics

Transaction Type	Statistic	Value
New Order	Rolledback transactions	1.00%
	Home warehouse	99.00%
	Remote warehouse	1.00%
	Average Items per Order	10.00
Payment	Home warehouse	85.01%
	Remote warehouse	14.99%
	Non-primary key access	60.02%
Order Status	Non-primary key access	59.98%
Delivery	Skipped transactions (Interactive)	0
	Skipped transaction counts (Deferred)	0
	Skipped District counts (Deferred)	0
Transaction Mix	New Order	44.81%
	Payment	43.07%
	Order-Status	4.03%
	Delivery	4.03%
	Stock-Level	4.03%

5.13. Checkpoint Statistics

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. (8.1.6.21)

There are four checkpoints in the measurement interval. The first checkpoint starts approximately 13 seconds into the measurement interval. The checkpoint interval is 30 minutes (from the start of one to the start of the next) and a checkpoint lasts approximately 29.8 minutes.

The start time and duration in seconds of at least the four (4) longest checkpoints during the Measurement Interval must be disclosed (see Clause 5.5.2.2 (2)). (8.1.6.11)

Table 5.5: Checkpoint Start Time and Duration

Item	Start	End	Duration
Measurement Interval	0:54:02	2:54:02	7200
1st Checkpoint	0:54:15	1:24:01	1786
2nd Checkpoint	1:24:14	1:54:00	1786
3rd Checkpoint	1:54:13	2:23:59	1786
4th Checkpoint	2:24:12	2:53:58	1786

6. Clause 6: SUT, Driver & Communications Definition

6.1. Remote Terminal Emulator (RTE) Description

The RTE input parameters, code fragments, functions, etc. used to generate each transaction input field must be disclosed. (8.1.7.1)

The RTE used is proprietary to Unisys. Appendix D contains the profile used as input to this RTE.

6.2. Loss of Terminal Connections

The number of terminal connections lost during the Measurement Interval must be disclosed (see Clause 6.6.2). (8.1.7.2)

No terminal connections were lost.

6.3. Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to that of the priced system. The results of the test described in Clause 6.6.3.4 must be disclosed. (8.1.7.3)

There were no emulated components in the benchmark configuration other than the emulated web browsers on the users' PCs.

6.4. Functional Diagrams

A complete functional diagram of both benchmark 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 (see Clause 6.6.3.6). (8.1.7.4)

Section 0.7 describes and shows functional diagrams of the benchmarked and priced systems.

6.5. Network Configuration

The network configuration of both the tested and proposed (target) services which are being represented and a thorough explanation of exactly which parts are being replaced with the Driver System must be disclosed (see Clause 6.6.4). (8.1.7.5)

Figures 0.1 and 0.2 in Section 0.7 also diagram the network configurations of the benchmark and configured systems and represent the RTEs connected via LAN replacing the user PCs that are directly connected via LAN.

6.6. Network Bandwidth

The bandwidth of the network(s) used in the tested/priced configuration must be disclosed. (8.1.7.6)

Local area networks (LANs) are used in the priced and tested configurations. The database server contains two cLAN 1.25 gigabit-per-second LAN adapters. Each was connected to a cLAN switch which was in turn connected in one case to four clients and in the other case five, each containing one cLAN adapter. These LAN segments run at 1.25 gigabit-per-second in the priced and tested configurations. Each client also contains one 10/100 megabit-per-second inbuilt LAN controller. Two sixteen-port 10/100 megabit-per-second switches per client were concatenated and used to connect the client to RTE driver systems emulating web browsers in the tested configuration. The 10/100 LAN connections between the clients and switches run at 100 megabits-per-second in the priced and tested configurations. 144 (16 per client) user LAN segments run at 10 megabits-per-second in the tested configuration.

6.7. Operator Intervention

If the configuration requires operator intervention (see Clause 6.6.6), the mechanism and the frequency of this intervention must be disclosed. (8.1.7.7)

No operator intervention was required to sustain eight hours of operation at the reported throughput.

7.1. 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 date. 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(s) and effective date(s) must also be reported. (8.1.8.1)

The total 3-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. (8.1.8.2)

System pricing should include subtotals for the following components: Server Hardware, Server Software, Client Hardware, Client Software, and Network Components used for terminal connection (see Clause 7.2.2.3). Clause 6.1 describes the Server and Client components. (8.1.8.7)

System pricing must include line item indication where non-sponsoring companies' brands are used. System pricing must also include line item indication of third party pricing. (8.1.8.8)

A detailed list of hardware and software components along with their part numbers and prices are given in the Executive Summary near the beginning of this document.

7.1.1. System Pricing

Each priced configuration consists of an integrated system package, additional options, and components. Prices for all products are US list prices. A three-year warranty is standard with this class of Unisys server products.

7.1.2. Maintenance Pricing

Three-year support pricing for Unisys Corporation products is based on standard new product warranty, upgraded to service level Comprehensive-Gold. Unisys's service level Comprehensive-Gold provides 24 hours/day, 7 days /week support with four hour maximum response for hardware spare parts.

Microsoft support pricing is based on 3 years of annual support costs.

Server disks are covered by Comark's 3 year, return-to-factory warranty with seven day maximum replenishment, and appropriate spares are included in the priced configuration.

7.1.3. Discounts

Unisys provides a standard pre-pay discount for maintenance service of the client, server and storage components of the priced configuration.

Comark provides a standard dollar-volume discount for the client, server and storage components of the priced configuration.

7.2. Availability

The committed delivery date for general availability (availability date) of products used in the price calculation must be reported. When the priced system includes 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. (8.1.8.3)

The hardware, software and support/maintenance products priced in this benchmark are detailed on page vi.

All components will be available by the date shown on pages v.

7.3. Measured tpmC, Pricing, Price/Performance, and Availability Date

A statement of the measured tpmC as well as the respective calculations for the 3-year pricing, price/performance (price/tpmC), and the availability date must be included. (8.1.8.4)

The measured tpmC, plus pricing calculations, price/performance, and availability are shown on pages v and vi.

7.4. 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. (8.1.8.5)

None.

7.5. Usage Pricing

For any usage pricing, the sponsor must disclose: (8.1.8.6)

- *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:

- One (1) Microsoft Windows Datacenter Server Limited Edition license
- One (1) Microsoft SQL Server 2000 Enterprise Edition license
- Nine (9) Microsoft Windows 2000 Server license
- One (1) Microsoft Visual C++ Professional 6.0 license

8.

Clause 8 : Full Disclosure Availability

8.1. Availability

The Full Disclosure Report must be readily available to the public at a reasonable charge, similar to charges for similar documents by that test sponsor. (8.2)

Copies of this Full Disclosure Report may be downloaded from the Transaction Processing Performance Council web site at www.tpc.org or obtained by contacting:

TPC Benchmark Administrator
Systems Analysis, Modeling & Measurement Group
Unisys Corporation, M/S 262
25725 Jeronimo Road
Mission Viejo, CA 92691
USA

9.1. 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. (8.1.9.1)

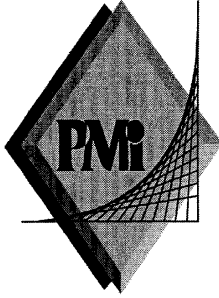
A review of the pricing model is required to ensure that all components required are priced (see Clause 9.2.8). The auditor is not required to review the final Full Disclosure Report or the final pricing prior to issuing the attestations letter. (8.1.9.2)

This implementation of the TPC Benchmark C on the Unisys e-@ction Enterprise Server ES7000 was audited by Lorna Livingtree, a TPC certified auditor of:

Performance Metrics, Inc.
137 Yankton St. Suite 101
Folsom, CA 95630

Phone: (916) 985-1131
Fax: (916) 985-1185
email: Lorna@PerfMetrics.com

The attestation letter is shown on the next 2 pages.



PERFORMANCE METRICS INC.
TPC Certified Auditors

August 1, 2001

Jerrold Buggert
Director of Modeling and Measurement
Unisys Corporation
25725 Jeronimo Road
Mission Viejo, CA 92691

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

Platform: Unisys e-@ction Enterprise Server ES7000
Database Manager: Microsoft SQL Server 2000 Enterprise Edition
Operating System: Microsoft Windows Datacenter Server Limited Edition
Transaction Manager: Microsoft COM+ (included in Windows 2000)

Server: e-@ction Enterprise Server ES7000				
CPU's	Memory	Disks	90% Response	TpmC
32 Pentium III Xeon @ 900 MHz	Main: 64 GB Cache: 2MB each	921 @ 18GB 24 @ 36GB	1.03 sec	141,138.44
9 Clients: e-@ction ES2024				
2 Pentium III @ 1 GHz	Main: 512 MB Cache: 256K	1 @ 18GB	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 was properly sized and populated.
- The database was properly scaled with 11,808 warehouses of which only 11,520 were active during the Measured Interval.
- The ACID properties were met.

PERFORMANCE METRICS INC.
TPC Certified Auditors

- The durability data loss and log loss tests were performed on a fully scaled database with 14,400 users active.
- Input data was generated according to the specified percentages.
- Eight hours of mirrored log space was configured on the measured system.
- Eight hours of dynamic table growth space was configured on the measured system.
- The 60-day space calculation was verified. The measured configuration has sufficient storage to satisfy this requirement.
- Measurement cycle times included a 0.1-second menu and a 0.1 second response time delay for an emulated Web browser.
- There were 115,200 user contexts present on the system.
- Each emulated user started with a different random number seed.
- The NURand constants used for database load and at run time were 123 and 208.
- The steady state portion of the test was 2 hours (7200 seconds).
- One checkpoint was taken before the measured interval.
- Four checkpoints were taken during the measured interval.
- The system pricing was checked for major components and maintenance.

Auditor Notes:

None.

Sincerely,



Lorna Livingtree
Auditor

Appendix A - Client Source

tpcc.def

```
EXPORTS
    GetExtensionVersion
    HttpExtensionProc
```

tpcc.h

```
// tpcc.h
//
// Copyright Unisys, 2000

#include <time.h>

#define VERSIONINFO "--- 1.2.8 ---"

// TPCCHandler return codes
#define TPCCSEND 1
#define TPCCSENDEND 2
#define TPCCENDNOW 3

// TPCC Service return codes
#define SVC_BADITEMID 1
#define SVC_NOERROR 0
#define SVCERR_DEADLOCK -1
#define SVCERR_NOCUSTOMER -2
#define SVCERR_NOORDERS -3
#define SVCERR_DBLIB -4
#define SVCERR_EXCEPTION -5
#define SVCERR_DQFULL -6
#define SVCERR_DQSTART -7
#define SVCERR_ODBC -8

// Min/Max transaction data definitions
#define MIN_Did 1
#define MAX_Did 10
#define MIN_OL 5
#define MAX_OL 15
#define MIN_QUANTITY 1
#define MAX_QUANTITY 10
#define MIN_ITEM_ID 1
#define MAX_ITEM_ID 100000
#define MIN_CUST_ID 1
#define MAX_CUST_ID 3000
#define MIN_CARRIER 1
#define MAX_CARRIER 10
#define MIN_THRESHOLD 10
#define MAX_THRESHOLD 20

// pTPCC->iStatusId codes
#define INVALID_IID 1
#define STATUS_OK 0
```

```
#define ERR_CMD_UNKNOWN -10
#define ERRTXT_CMD_UNKNOWN "Unrecognized Command"
#define ERR_ALREADY_LOGGEDIN -11
#define ERRTXT_ALREADY_LOGGEDIN "Already Logged In"
#define ERR_TERMID -12
#define ERRTXT_TERMID "TermId or SyncId in Error"
#define ERR_FORM_UNKNOWN -13
#define ERRTXT_FORM_UNKNOWN "Unrecognized FormId"
#define ERR_WID_INVALID -14
#define ERR_DID_INVALID -15
#define ERR_MISSING_KEY -16
#define ERR_NOT_NUMERIC -17
#define ERR_THRESHOLD_RANGE -18
#define ERR_EMBEDDED_EMPTY_OL -19
#define ERR_QUANTITY_INVALID -20
#define ERR_OL_INVALID -21
#define ERR_OL_COUNT -22
#define ERR_TM_INTERFACE -23
#define ERR_SERVICE_RSLT -24
#define ERR_INPUT_TOOLONG -25
#define ERR_IDANDNAME_EMPTY -26
#define ERR_IDANDNAME_ENTERED -27
#define ERR_AMOUNT_BADFORM -28
#define ERR_AMOUNT_INVALID -29
#define ERR_CARRIER_INVALID -30
#define ERR_TERM_ALLOC -31

#define STATUS_LEN 30
#define NAME_LEN 16
#define ADDR_LEN 20
#define STATE_LEN 2
#define ZIP_LEN 9

#define CTEXT "Content-length: "
#define HTTPHdr "Content-type: text/html\r\nContent-length: \r\n\r\n"

typedef struct
{
    unsigned short year;
    unsigned short month;
    unsigned short day;
    unsigned short hour;
    unsigned short minute;
    unsigned short second;
} DATEDetails;

typedef struct
{
    short ol_supply_w_id;
    long ol_i_id;
    char ol_i_name[25];
    short ol_quantity;
    char ol_brand_generic[2];
    double ol_i_price;
    double ol_amount;
    short ol_stock;
```

```

} OL_NEW_ORDER_DATA;

typedef struct
{
    bool bTPRslt;
    short iTPRslt;
    short w_id;
    short d_id;
    long c_id;
    short o_ol_cnt;
    char c_last[NAME_LEN + 1];
    char c_credit[3];
    double c_discount;
    double w_tax;
    double d_tax;
    long o_id;
    short o_all_local;
    short o_commit_flag;
    DATEDETAILS o_entry_d;
    double total_amount;
    char execution_status[STATUS_LEN];
    OL_NEW_ORDER_DATA ol[MAX_OL];
} NEW_ORDER_DATA;

typedef struct
{
    bool bTPRslt;
    short iTPRslt;
    short w_id;
    short d_id;
    long c_id;
    short c_d_id;
    short c_w_id;
    double h_amount;
    DATEDETAILS h_date;
    char w_street_1[ADDR_LEN + 1];
    char w_street_2[ADDR_LEN + 1];
    char w_city[ADDR_LEN + 1];
    char w_state[STATE_LEN + 1];
    char w_zip[ZIP_LEN + 1];
    char d_street_1[ADDR_LEN + 1];
    char d_street_2[ADDR_LEN + 1];
    char d_city[ADDR_LEN + 1];
    char d_state[STATE_LEN + 1];
    char d_zip[ZIP_LEN + 1];
    char c_first[NAME_LEN + 1];
    char c_middle[3];
    char c_last[NAME_LEN + 1];
    char c_street_1[ADDR_LEN + 1];
    char c_street_2[ADDR_LEN + 1];
    char c_city[ADDR_LEN + 1];
    char c_state[STATE_LEN + 1];
    char c_zip[ZIP_LEN + 1];
    char c_phone[17];
    DATEDETAILS c_since;
    char c_credit[3];
    double c_credit_lim;
    double c_discount;
    double c_balance;
    char c_data[200+1];
    char execution_status[STATUS_LEN];

```

```

} PAYMENT_DATA;

typedef struct
{
    long ol_i_id;
    short ol_supply_w_id;
    short ol_quantity;
    double ol_amount;
    DATEDETAILS ol_delivery_d;
} OL_ORDER_STATUS_DATA;

typedef struct
{
    bool bTPRslt;
    short iTPRslt;
    short w_id;
    short d_id;
    long c_id;
    char c_first[NAME_LEN + 1];
    short o_ol_cnt;
    char c_middle[3];
    char c_last[NAME_LEN + 1];
    double c_balance;
    long o_id;
    DATEDETAILS o_entry_d;
    short o_carrier_id;
    OL_ORDER_STATUS_DATA olOrderStatusData[MAX_OL];
    char execution_status[STATUS_LEN];
} ORDER_STATUS_DATA;

typedef struct
{
    bool bTPRslt;
    short iTPRslt;
    short w_id;
    short o_carrier_id;
    long o_id[10];
    SYSTEMTIME QTime; // time delivery was queued
    SYSTEMTIME EndTime; // time delivery completed
} DELIVERY_DATA;

typedef struct
{
    bool bTPRslt;
    short iTPRslt;
    short w_id;
    short d_id;
    short thresh_hold;
    long low_stock;
    char execution_status[STATUS_LEN];
} STOCK_LEVEL_DATA;

// tpcc.cpp
//
// Copyright Unisys, 1999
//

```

tpcc.cpp

```

#include <windows.h>
#include <stdio.h>
#include <malloc.h>
#include <stdlib.h>
#include <string.h>
#include <winreg.h>
#include <httpext.h>

#include "..\tpccsvr\tpcc.h"
#include "tmon.h"
#include "diagio.h"
#include "term.h"
#include "delivery.h"
#include "tpcchandler.h"

#define EXTN_VERSION MAKELONG(HSE_VERSION_MINOR,HSE_VERSION_MAJOR)
#define TLS_NULL 0xFFFFFFFF
DWORD dwTlsInx;
CHAR * pTitle = "IIS TPCC COM DLL";
CRITICAL_SECTION csDllMain;

// Diagnostic logging settings
BOOL bSetEventLog = TRUE;
BOOL bSetConsole = FALSE;
UINT uSetDiagLevel = DIAG_INFO;

// TMon Interface Settings
INT iTMaxMsg = 0;

// Term Interface Settings
INT iMaxTerms = 3000;

// Delivery Settings
long lSetDThreads = 8;
long lSetDQSize = DEFAULTDQSIZE;
char szSetPath[200] = "\\inetpub\\wwwroot\\";

static CHAR * szTPCCError =
    HTTPHdr "<HTML>"
    "<HEAD><TITLE>Welcome To TPC-C</TITLE></HEAD><BODY>"
    "<B>TPCC Extension Error (TPCC Array Not Allocated)</B><BR>"
    "</BODY></HTML>";

static CHAR * szTMinInitError =
    HTTPHdr "<HTML>"
    "<HEAD><TITLE>Welcome To TPC-C</TITLE></HEAD><BODY>"
    "<B>TPCC Extension Error (TMinInit Failed)</B><BR>"
    "</BODY></HTML>";
INT iHHdrLen = 0;
INT iCTextLen = 0;

BOOL ThreadAttach(TPCC_STATE * pTPCC,CHAR * pDiag);
VOID ThreadDetach(TPCC_STATE * pTPCC);
VOID SendResponse(EXTENSION_CONTROL_BLOCK * pECB,CHAR * pMsg,CHAR *
pWork);
BOOL ReadRegistry(VOID);

//=====
//
// Function name: DllMain
//

```

```

//=====
BOOL APIENTRY DllMain(HANDLE hInst, ULONG ul_reason_for_call,
LPVOID lpReserved)
{
    TPCC_STATE * pTPCC = NULL;
    CHAR szDiag[MAX_DIAG_SZ];
    UINT iTMaxSz = 0;
    switch(ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
            // Process initialization

            InitializeCriticalSection(&csDllMain);
            ReadRegistry();
            DiagIoInit(pTitle,bSetConsole,bSetEventLog,uSetDiagLevel);
            wsprintf(szDiag,
                "(%s) EventLog = %d, Console = %d, DiagLevel = %d\n"
                "MaxTerms = %d\n",
                VERSIONINFO,bSetEventLog,bSetConsole,uSetDiagLevel,iMaxTerms);
            DiagIoWrite(szDiag,DIAG_FORCE);
            dwTlsInx = TlsAlloc();
            if (dwTlsInx == TLS_NULL)
            {
                wsprintf(szDiag,"PAttach(%ld): Tls Alloc Failed (%ld)\n",
                    GetCurrentThreadId(),GetLastError());
                DiagIoWrite(szDiag,DIAG_ERROR);
                return(FALSE);
            };
            if (TermInit(iMaxTerms))
                return(FALSE);
            iTMaxSz = max(iTMaxSz,sizeof(NEW_ORDER_DATA));
            iTMaxSz = max(iTMaxSz,sizeof(PAYMENT_DATA));
            iTMaxSz = max(iTMaxSz,sizeof(ORDER_STATUS_DATA));
            iTMaxSz = max(iTMaxSz,sizeof(DELIVERY_DATA));
            iTMaxSz = max(iTMaxSz,sizeof(STOCK_LEVEL_DATA));
            iTMaxSz += 10;
            TMonInit(iTMaxSz);
            if (DeliveryInit(lSetDThreads,lSetDQSize,szSetPath))
            {
                DeliveryTerm();
                return(FALSE);
            };
            iHHdrLen = strlen(HTTPHdr);
            iCTextLen = strlen(CTEXT);
            break;
        case DLL_THREAD_ATTACH:
            // Move ThreadAttach call to HttpExt since the DllMain call
            // for Thread Attach did not reliably come before the first
            // call to HttpExtProc.
            break;
        case DLL_THREAD_DETACH:
            ThreadDetach(pTPCC);
            break;
        case DLL_PROCESS_DETACH:
            ThreadDetach(pTPCC);
            DeleteCriticalSection(&csDllMain);
            DeliveryTerm();
            TMonTerm();
            TermTerm();
            TlsFree(dwTlsInx);
            dwTlsInx = TLS_NULL;
    }
}

```

```

        DiagIoTerm();
                break;
    };
    return TRUE;
}; // DllMain

//=====
//
// Function name: ThreadAttach
//
// Result:
// FALSE Thread state structure initialized
// TRUE Thread state structure initialization failure
//
//=====
BOOL ThreadAttach(TPCC_STATE * pTPCC, CHAR * pDiag)
{
    EnterCriticalSection(&csDllMain);
    __try
    {
        pTPCC = (TPCC_STATE *) calloc(1, sizeof(TPCC_STATE));
        if (pTPCC == NULL)
        {
            wsprintf(pDiag, "ThrAtt(%ld): pTPCC Alloc Failed (%ld)\n",
                GetCurrentThreadId(), GetLastError());
            DiagIoWrite(pDiag, DIAG_ERROR);
            return(TRUE);
        };
        TlsSetValue(dwTlsInx, pTPCC);
        pTPCC->tsTMon.pszErrTxt = pTPCC->ErrTxt;
        if (TMinInit(&pTPCC->tsTMon))
        {
            wsprintf(pDiag, "ThrAtt(%ld): TMinInit %s\n",
                GetCurrentThreadId(), pTPCC->ErrTxt);
            DiagIoWrite(pDiag, DIAG_ERROR);
            return(TRUE);
        };
    }
    __finally
    {
        LeaveCriticalSection(&csDllMain);
    };
    return(FALSE);
}; // ThreadAttach

//=====
//
// Function name: ThreadDetach
//
//=====
VOID ThreadDetach(TPCC_STATE * pTPCC)
{
    EnterCriticalSection(&csDllMain);
    __try
    {
        pTPCC = (TPCC_STATE *) TlsGetValue(dwTlsInx);
        if (pTPCC != NULL)
        {
            TMDone(&pTPCC->tsTMon);
            free(pTPCC);
            pTPCC = NULL;

```

```

        TlsSetValue(dwTlsInx, pTPCC);
    };
}
__finally
{
    LeaveCriticalSection(&csDllMain);
}; // ThreadDetach

//=====
//
// Function name: GetExtensionVersion
//
//=====
BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pVersion)
{
    pVersion->dwExtensionVersion = EXTN_VERSION;
    strncpy(pVersion->lpszExtensionDesc, pTitle, HSE_MAX_EXT_DLL_NAME_LEN);
    return TRUE;
}; // GetExtensionVersion

//=====
//
// Function name: HttpExtensionProc
//
// Returns:
// HSE_STATUS_SUCCESS send msg, drop connection
// HSE_STATUS_SUCCESS_AND_KEEP_CONN send msg, keep connection
//
//=====
DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK * pECB)
{
    TPCC_STATE * pTPCC;
    DWORD dwRslt = HSE_STATUS_SUCCESS;
    UINT uRslt;

    pTPCC = (TPCC_STATE *) TlsGetValue(dwTlsInx);
    if (pTPCC == NULL)
    {
        CHAR szWork[200];
        ThreadAttach(pTPCC, szWork);
        pTPCC = (TPCC_STATE *) TlsGetValue(dwTlsInx);
        if (pTPCC == NULL)
        {
            SendResponse(pECB, szTPCCError, szWork);
            goto HttpExit;
        };
    };
    if (pTPCC->tsTMon.pTxnData == NULL)
        SendResponse(pECB, szTMinInitError, pTPCC->szHeader);
    TPCCclear(pTPCC);
    pTPCC->ConnID = pECB->ConnID;
    pTPCC->RecvMsg = pECB->lpszQueryString;
    uRslt = TPCCHandler(pTPCC);
    switch (uRslt)
    {
        case TPCCSEND:
            SendResponse(pECB, pTPCC->SendMsg, pTPCC->szHeader);
            dwRslt = HSE_STATUS_SUCCESS_AND_KEEP_CONN;
            break;

```

```

    case TPCCSENDEND:
        SendResponse(pECB,pTPCC->SendMsg,pTPCC->szHeader);
        break;
    case TPCCENDNOW:
    default:
        break;
}; // switch (TPCCHandler result)

HttpXit:

    return(dwRslt);

}; // HttpExtensionProc

//=====
//
// Function name: SendResponse
//
//=====
VOID SendResponse(EXTENSION_CONTROL_BLOCK * pECB,CHAR * pMsg,CHAR * pWork)
{
    DWORD dwMsgBytes;
    DWORD dwDataBytes;
    CHAR * pCL;
    HSE_SEND_HEADER_EX_INFO HeaderExInfo;
    dwMsgBytes = strlen(pMsg);
    pCL=strstr(pMsg,CTEXT);
    dwDataBytes = dwMsgBytes - iHHdrLen;
    wsprintf(pWork,"%4ld",dwDataBytes);
    pCL += iCTextLen;
    strncpy(pCL,pWork,4);
    HeaderExInfo.pszHeader = pMsg;
    HeaderExInfo.cchHeader = dwMsgBytes;
    HeaderExInfo.pszStatus = "200 OK";
    HeaderExInfo.cchStatus = 6;
    HeaderExInfo.fKeepConn = TRUE;
    (*pECB->ServerSupportFunction)
        (pECB->ConnID,
         HSE_REQ_SEND_RESPONSE_HEADER_EX,
         &HeaderExInfo,
         NULL,
         NULL);
}; // SendResponse

//=====
//
// Function name: ReadRegistry
//
// Sets global operational parameters from registry if they exist.
// Otherwise, compiled in defaults apply.
//
// Result:
// FALSE Registry entry found
// TRUE Registry entry does not exist
//
//=====
BOOL ReadRegistry(VOID)
{
    HKEY hkTPCC;
    DWORD dwMax;
    DWORD dwRT;

```

```

    INT i;
    CHAR szValue[100];
    if (RegOpenKeyEx(HKEY_LOCAL_MACHINE,"SOFTWARE\\Unisys\\TPCC",0,
        KEY_READ, &hkTPCC) != ERROR_SUCCESS )
        return(TRUE);
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"EVENTLOG",0,&dwRT,(BYTE *) &szValue,&dwMax)
        == ERROR_SUCCESS)
    {
        if (abs(atoi(szValue) == 0))
            bSetEventLog = FALSE;
        else
            bSetEventLog = TRUE;
    };
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"CONSOLE",0,&dwRT,(BYTE *) &szValue,&dwMax)
        == ERROR_SUCCESS )
    {
        if (abs(atoi(szValue) == 0))
            bSetConsole = FALSE;
        else
            bSetConsole = TRUE;
    };
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"DIAGLEVEL",0,&dwRT,(BYTE *)
&szValue,&dwMax)
        == ERROR_SUCCESS )
    {
        i = atoi(szValue);
        if (i < DIAG_FORCE)
            i = DIAG_FORCE;
        else
            if (i > DIAG_INFO)
                i = DIAG_INFO;
        uSetDiagLevel = i;
    };
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"MAXTERMS",0,&dwRT,(BYTE *) &szValue,&dwMax)
        == ERROR_SUCCESS )
    {
        iMaxTerms = abs(atoi(szValue));
    };
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"DELIVERYTHREADS",0,&dwRT,(BYTE *)
&szValue,&dwMax)
        == ERROR_SUCCESS )
    {
        lSetDThreads = abs(atoi(szValue));
    };
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"DQSIZE",0,&dwRT,(BYTE *) &szValue,&dwMax)
        == ERROR_SUCCESS )
    {
        lSetDQSize = abs(atoi(szValue));
    };
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"DQPATH",0,&dwRT,(BYTE *) &szValue,&dwMax)
        == ERROR_SUCCESS )
    {
        strcpy(szSetPath,szValue);
    };

```

```

    RegCloseKey(hkTPCC);
    return(FALSE);
}; // ReadRegistry

```

tpcchandler.h

```

// tpcchandler.h
//
// Copyright Unisys, 1999
#define MAX_MSG_SZ 5000

typedef struct
{
    LPVOID ConnID; // Active Connection Id
    SHORT sWId; // TPCC WareHouse Id
    SHORT sDId; // TPCC District Id
    INT iSyncId; // TPCC Sync Id
    INT iTermId; // TPCC Term Id
    UINT uFormId; // TPCC Form Id
    INT iStatusId; // TPCC Status Id
    CHAR ErrTxt[500]; // Error text
    CHAR szWork[200]; // Thread work area
    CHAR szHeader[100]; // HTTP work area
    CHAR * RecvMsg; // HTML message from ECB
    CHAR SendMsg[MAX_MSG_SZ]; // HTML work area
    TMON_STATE tsTMon; // TMon Interface
} TPCC_STATE;

BOOL TPCCclear(TPCC_STATE * pTPCC);
UINT TPCCHandler(TPCC_STATE * pTPCC);

```

tpcchandler.cpp

```

// tpcchandler.cpp
//
// Copyright Unisys, 1999
//
#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#include "..\tpccsvr\tpcc.h"
#include "tmon.h"
#include "diagio.h"
#include "delivery.h"
#include "tpcchandler.h"
#include "term.h"

// pTPCC->iFormId - TPCC forms enumeration.
#define FORM_NULL 0
#define FORM_LOGON 1
#define FORM_MENU 2
#define FORM_NEWORDER 3
#define FORM_PAYMENT 4

```

```

#define FORM_DELIVERY 5
#define FORM_ORDERSTATUS 6
#define FORM_STOCKLEVEL 7
#define FORM_EXIT 8
#define FORM_MAX 9

// CMD= HTML Command Enumeration and Name
#define CMD_NULL 0
#define CMD_PROCESS 1
#define CMD_NEWORDER_FORM 2
#define CMD_PAYMENT_FORM 3
#define CMD_DELIVERY_FORM 4
#define CMD_ORDERSTATUS_FORM 5
#define CMD_STOCKLEVEL_FORM 6
#define CMD_EXIT 7
#define CMD_SUBMIT 8
#define CMD_MENU_FORM 9
#define CMD_MAX 10

static CHAR * szCmds[] =
{
    "Unknown",
    "Process",
    "..NewOrder..",
    "..Payment..",
    "..Delivery..",
    "..Order-Status..",
    "..Stock-Level..",
    "..Exit..",
    "Submit",
    "Menu"
};

static CHAR * szFormLogin =
    HTTPHdr "<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=\"FORMID\" VALUE=\"1\">"
    "<INPUT TYPE=\"hidden\" NAME=\"TERMINID\" 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>";

static CHAR * szMenuList =
    "<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..\">";

static CHAR * HTMLTrailer =
    "</BODY></HTML>";

static CHAR * TERMINIDTOKEN = "TERMINID=";
static CHAR * SYNCIDTOKEN = "SYNCID=";

```

```

static CHAR * FORMIDTOKEN = "FORMID=";
static CHAR * STATUSIDTOKEN = "STATUSID=";
static CHAR * CMDTOKEN = "CMD=";
static CHAR * NEWORDER_SERVICE = "NEWORDER";
static CHAR * PAYMENT_SERVICE = "PAYMENT";
static CHAR * ORDERSTATUS_SERVICE = "ORDERSTS";
static CHAR * DELIVERY_SERVICE = "DELIVERY";
static CHAR * STOCKLEVEL_SERVICE = "STOCKLVL";
static CHAR * ZIPPIC = "XXXXX-XXXX";

BOOL ProcessLogin(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC);
BOOL ProcessForm(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC);
BOOL ProcessNewOrder(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC);
BOOL ProcessPayment(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC);
BOOL ProcessDelivery(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC);
BOOL ProcessOrderStatus(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC);
BOOL ProcessStockLevel(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC);
VOID FormatLogin(CHAR * pMsg,CHAR * pAddText);
BOOL GetHidden(CHAR * pMsg,UINT * uFormId,INT * iSyncId,INT * iTermId);
BOOL GetCmd(CHAR * pMsg,CHAR * pWork,UINT uLen);
BOOL GetLongKey(LONG * lRslt,CHAR * pHTML,CHAR * pKey,TPCC_STATE * pTPCC);
BOOL GetIntKey(INT * iRslt,CHAR * pHTML,CHAR * pKey,TPCC_STATE * pTPCC);
BOOL GetShortKey(SHORT * sRslt,CHAR * pHTML,CHAR * pKey,TPCC_STATE *
pTPCC);
BOOL GetStringKey(CHAR * szRslt,CHAR * pHTML,CHAR * pKey,
TPCC_STATE * pTPCC,UINT uMax);
BOOL GetAmountKey(DOUBLE * dRslt,CHAR * pHTML,CHAR * pKey,
TPCC_STATE * pTPCC);
BOOL GetKeyValue(CHAR * pHTML,CHAR * pKey,CHAR * pValue,UINT uMax);
VOID FormatLogin(CHAR * pOut,CHAR * pAddText);
VOID FormatMenu(CHAR * pOut,TPCC_STATE * pTPCC);
VOID FormatNewOrder(CHAR * pOut,TPCC_STATE * pTPCC);
VOID FormatPayment(CHAR * pOut,TPCC_STATE * pTPCC);
VOID FormatDelivery(CHAR * pOut,TPCC_STATE * pTPCC);
VOID FormatOrderStatus(CHAR * pOut,TPCC_STATE * pTPCC);
VOID FormatStockLevel(CHAR * pOut,TPCC_STATE * pTPCC);
INT FormatFormHdr(CHAR * pOut,CHAR * pTitle,TPCC_STATE * pTPCC);
INT FormatRespHdr(CHAR * pOut,CHAR * pTitle,TPCC_STATE * pTPCC);
VOID FormatString(CHAR * pOut,CHAR * pPic,CHAR * pIn);
VOID UtilStrCpy(CHAR * pDest,CHAR * pSrc,INT n);
BOOL CheckNumeric(CHAR * pNum);

//=====
//
// Function name: TPCCclear
//
//=====
BOOL TPCCclear(TPCC_STATE * pTPCC)
{
    pTPCC->ConnID = 0;
    pTPCC->sWId = 0;
    pTPCC->sDId = 0;
    pTPCC->iSyncId = 0;
    pTPCC->iTermId = -2;
    pTPCC->uFormId = FORM_NULL;
    pTPCC->iStatusId = 0;
    strcpy(pTPCC->ErrMsg, "");
    return(FALSE);
}; // TPCCclear

//=====

```

```

//
// Function name: TPCCHandler
//
//=====
UINT TPCCHandler(TPCC_STATE * pTPCC)
{
    INT iSyncId;
    INT iTermId;
    UINT uCmdId;
    UINT uRslt = TPCCSEND; // default error handling
    TERM_STATE * pTerm;

    pTPCC->iStatusId = STATUS_OK;
    if (GetHidden(pTPCC->RecvMsg,&pTPCC->uFormId,&iSyncId,&iTermId))
    {
        uRslt = TPCCSEND;
        FormatLogin(pTPCC->SendMsg,pTPCC->ErrMsg);
        goto HdlrXit;
    };
    if (iTermId > 0)
    {
        pTerm = TermGet(iTermId);
        if (pTerm == NULL)
        {
            uRslt = TPCCSEND;
            strcpy(pTPCC->ErrMsg,"Invalid Term Id");
            FormatLogin(pTPCC->SendMsg,pTPCC->ErrMsg);
            goto HdlrXit;
        };
        pTPCC->sWId = pTerm->sWId;
        pTPCC->sDId = pTerm->sDId;
        pTPCC->iSyncId = pTerm->iSyncId;
        pTPCC->iTermId = pTerm->iTermId;
    };
    uCmdId = GetCmd(pTPCC->RecvMsg,pTPCC->szWork,sizeof(pTPCC->szWork));
    // Except for Submit(log in), sWId must already be set
    if (pTPCC->sWId == 0 && uCmdId != CMD_SUBMIT)
    {
        strcpy(pTPCC->ErrMsg,"Must log in first!");
        FormatLogin(pTPCC->SendMsg,pTPCC->ErrMsg);
        uRslt = TPCCSEND;
        goto HdlrXit;
    };
    // Check for multiple log in attempts
    if (pTPCC->sWId != 0 && uCmdId == CMD_SUBMIT)
    {
        strcpy(pTPCC->ErrMsg,ERRTXT_ALREADY_LOGGEDIN);
        pTPCC->iStatusId = ERR_ALREADY_LOGGEDIN;
        FormatMenu(pTPCC->SendMsg,pTPCC);
        uRslt = TPCCSEND;
        goto HdlrXit;
    };
    // If not logging in, validate hidden fields
    if (uCmdId != CMD_SUBMIT)
    {
        if (iTermId != pTPCC->iTermId || iTermId != iSyncId)
        {
            wsprintf(pTPCC->ErrMsg,"%s: Received %ld, %ld (%ld)",
                ERRTXT_TERMID,iTermId,iSyncId,pTPCC->iTermId);
            pTPCC->iStatusId = ERR_TERMID;
            FormatMenu(pTPCC->SendMsg,pTPCC);
        };
    };
}

```

```

        goto HdlrXit;
    };
};
// Process the command
switch (uCmdId)
{
    case CMD_SUBMIT:
        ProcessLogin(pTPCC->RecvMsg, pTPCC->SendMsg, pTPCC);
        break;
    case CMD_MENU_FORM:
        FormatMenu(pTPCC->SendMsg, pTPCC);
        break;
    case CMD_PROCESS:
        ProcessForm(pTPCC->RecvMsg, pTPCC->SendMsg, pTPCC);
        break;
    case CMD_NEWORDER_FORM:
        FormatNewOrder(pTPCC->SendMsg, pTPCC);
        break;
    case CMD_PAYMENT_FORM:
        FormatPayment(pTPCC->SendMsg, pTPCC);
        break;
    case CMD_DELIVERY_FORM:
        FormatDelivery(pTPCC->SendMsg, pTPCC);
        break;
    case CMD_ORDERSTATUS_FORM:
        FormatOrderStatus(pTPCC->SendMsg, pTPCC);
        break;
    case CMD_STOCKLEVEL_FORM:
        FormatStockLevel(pTPCC->SendMsg, pTPCC);
        break;
    case CMD_EXIT:
        TermFree(pTPCC->iTermId);
        strcpy(pTPCC->ErrTxt, "Logged Off");
        FormatLogin(pTPCC->SendMsg, pTPCC->ErrTxt);
        goto HdlrXit;
    default:
        strcpy(pTPCC->ErrTxt, ERRTXT_CMD_UNKNOWN);
        pTPCC->iStatusId = ERR_CMD_UNKNOWN;
        if (pTPCC->sWid == 0)
            FormatLogin(pTPCC->SendMsg, pTPCC->ErrTxt);
        else
            FormatMenu(pTPCC->SendMsg, pTPCC);
        break;
}; // switch (uCmdId)

uRslt = TPCCSEND;

HdlrXit:

    return(uRslt);
}; // TPCCHandler

//=====
//
// Function name: ProcessLogin
//
// ProcessLogin extracts WId and DId from the incoming form. Assumes
// log in has not previously completed (sWid == 0 already verified).
//
// Result:

```

```

// FALSE - log in successful, sWid and sDId set in pTPCC,
//         pOut contains menu.
// TRUE  - log in failed, pOut contains log in form with
//         error message.
//
//=====
BOOL ProcessLogin(CHAR * pIn, CHAR * pOut, TPCC_STATE * pTPCC)
{
    SHORT sWid;
    SHORT sDId;
    TERM_STATE * pTerm;

    if (GetShortKey(&sWid, pIn, "w_id", pTPCC))
    {
        FormatLogin(pOut, pTPCC->ErrTxt);
        return(TRUE);
    };
    if (sWid < 1)
    {
        wsprintf(pTPCC->ErrTxt, "Warehouse Id (%d) Invalid", sWid);
        pTPCC->iStatusId = ERR_WID_INVALID;
        FormatLogin(pOut, pTPCC->ErrTxt);
        return(TRUE);
    };
    if (GetShortKey(&sDId, pIn, "d_id", pTPCC))
    {
        FormatLogin(pOut, pTPCC->ErrTxt);
        return(TRUE);
    };
    if (sDId < MIN_DId || sDId > MAX_DId)
    {
        wsprintf(pTPCC->ErrTxt, "DId Out of Range(%ld,%ld) - %ld",
            MIN_DId, MAX_DId, sDId);
        pTPCC->iStatusId = ERR_DID_INVALID;
        FormatLogin(pOut, pTPCC->ErrTxt);
        return(TRUE);
    };
    pTerm = TermAlloc();
    if (pTerm == NULL)
    {
        wsprintf(pTPCC->ErrTxt, "Unable to Allocate Terminal Entry");
        pTPCC->iStatusId = ERR_TERM_ALLOC;
        FormatLogin(pOut, pTPCC->ErrTxt);
        return(TRUE);
    };
    pTerm->ConnID = pTPCC->ConnID;
    pTerm->iSyncId = pTerm->iTermId;
    pTerm->sWid = abs(sWid);
    pTerm->sDId = abs(sDId);
    pTPCC->iTermId = pTerm->iTermId;
    pTPCC->iSyncId = pTerm->iSyncId;
    pTPCC->sWid = pTerm->sWid;
    pTPCC->sDId = pTerm->sDId;
    FormatMenu(pOut, pTPCC);
    return(FALSE);
}; // ProcessLogin

//=====
//
// Function name: ProcessForm
//

```



```

// ProcessForm uses pTPCC->uFormId to determine which form input is
// present and ready for processing. Actual processing is done by
// the form specific routine.
//
// Result:
// FALSE - form processed, pOut contains response.
// TRUE - error processing form input, pOut contains reason.
//
//=====
BOOL ProcessForm(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC)
{
    switch (pTPCC->uFormId )
    {
        case FORM_NEWORDER:
            return(ProcessNewOrder(pIn,pOut,pTPCC));
        case FORM_PAYMENT:
            return(ProcessPayment(pIn,pOut,pTPCC));
        case FORM_DELIVERY:
            return(ProcessDelivery(pIn,pOut,pTPCC));
        case FORM_ORDERSTATUS:
            return(ProcessOrderStatus(pIn,pOut,pTPCC));
        case FORM_STOCKLEVEL:
            return(ProcessStockLevel(pIn,pOut,pTPCC));
        default:
            wsprintf(pTPCC->ErrTxt,"%s (%ld)",
                ERRTXT_FORM_UNKNOWN,pTPCC->uFormId);
            pTPCC->iStatusId = ERR_FORM_UNKNOWN;
            FormatMenu(pOut,pTPCC);
            break;
    }
    return(TRUE);
}; // ProcessForm

//=====
//
// Function name: ProcessNewOrder
//
// ProcessNewOrder extracts the input data fields from pIn, processes
// the data, and returns a response in pOut.
//
// Result:
// FALSE - NewOrder processed successfully.
// TRUE - NewOrder processing failed.
//
//=====
BOOL ProcessNewOrder(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC)
{
    NEW_ORDER_DATA * pnod;
    TMON_STATE * pTMon;
    CHAR szKey[20];
    CHAR * ptr;
    INT iInx;
    UINT u;
    UINT uLine;
    HRESULT hr;
    int iSize;

    pTMon = &pTPCC->tsTMon;
    pnod = (NEW_ORDER_DATA *) pTMon->pTxnData;
    ZeroMemory(pnod,sizeof(NEW_ORDER_DATA));
    pnod->w_id = pTPCC->swId;

```

```

if (GetShortKey(&pnod->d_id,pIn,"DID*",pTPCC))
{
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
if (pnod->d_id < MIN_DId || pnod->d_id > MAX_DId)
{
    wsprintf(pTPCC->ErrTxt,"Did Out of Range(%ld,%ld) - %ld",
        MIN_DId,MAX_DId,pnod->d_id);
    pTPCC->iStatusId = ERR_DID_INVALID;
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
if (GetLongKey(&pnod->c_id,pIn,"CID*",pTPCC))
{
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
pnod->o_ol_cnt = 0;
ptr = pIn;
for(u=0, uLine=0; u < MAX_OL; u++)
{
    wsprintf(szKey,"SP%2.2d*",u);
    ptr = strstr(ptr,szKey);
    if (GetShortKey(&pnod->Ol[uLine].ol_supply_w_id,ptr,szKey,pTPCC))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    wsprintf(szKey,"IID%2.2d*",u);
    if (GetLongKey(&pnod->Ol[uLine].ol_i_id,ptr,szKey,pTPCC))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    wsprintf(szKey,"Qty%2.2d*",u);
    if (GetShortKey(&pnod->Ol[uLine].ol_quantity,ptr,szKey,pTPCC))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    if (pnod->Ol[uLine].ol_i_id != 0)
    {
        if (pnod->Ol[uLine].ol_supply_w_id < 1)
        {
            wsprintf(pTPCC->ErrTxt,
                "Order Line %ld Contains Invalid Wid %d",
                u + 1,pnod->Ol[uLine].ol_supply_w_id);
            pTPCC->iStatusId = ERR_WID_INVALID;
            FormatMenu(pOut,pTPCC);
            return(TRUE);
        };
        if (pnod->Ol[uLine].ol_quantity < MIN_QUANTITY ||
            pnod->Ol[uLine].ol_quantity > MAX_QUANTITY)
        {
            wsprintf(pTPCC->ErrTxt,
                "Order Line %ld Contains Invalid Qty %d",
                u + 1,pnod->Ol[uLine].ol_quantity);
            pTPCC->iStatusId = ERR_QUANTITY_INVALID;
            FormatMenu(pOut,pTPCC);
            return(TRUE);
        };
    };
};

```

```

};
uLine++;
} // if (ol_i_id !=0)
else
{
if (pnod->Ol[uLine].ol_supply_w_id != 0)
{
wsprintf(pTPCC->ErrTxt,
"Order Line %ld WId Supplied with No Item",u + 1);
pTPCC->iStatusId = ERR_OL_INVALID;
FormatMenu(pOut,pTPCC);
return(TRUE);
};
if (pnod->Ol[uLine].ol_quantity != 0)
{
wsprintf(pTPCC->ErrTxt,
"Order Line %ld Qty Supplied with No Item",u + 1);
pTPCC->iStatusId = ERR_OL_INVALID;
FormatMenu(pOut,pTPCC);
return(TRUE);
};
}; // empty order line
}; // for (u < MAX_OL)
pnod->o_ol_cnt = uLine;
if (pnod->o_ol_cnt < MIN_OL)
{
wsprintf(pTPCC->ErrTxt,"Too Few Order Lines %d",pnod->o_ol_cnt);
pTPCC->iStatusId = ERR_OL_COUNT;
FormatMenu(pOut,pTPCC);
return(TRUE);
};
iSize = pTMon->iSize;
hr = pTMon->pIAllTxn->NewOrder(&iSize,(unsigned char*)&pTMon-
>pTxnData);
if (FAILED(hr))
{
pTPCC->iStatusId = ERR_TM_INTERFACE;
wsprintf(pTPCC->ErrTxt,
"COM Interface to NewOrder Call Failed, HRESULT %x",
hr);
FormatMenu(pOut,pTPCC);
return(TRUE);
};
};
pnod = (NEW_ORDER_DATA *) pTMon->pTxnData;
// Exclude invalid item id case
if (pnod->bTPRslt && pnod->iTPRslt < SVC_NOERROR)
{
wsprintf(pTPCC->ErrTxt,
"New Order Service Returned Error(%ld): %s",
pnod->iTPRslt,pnod->execution_status);
pTPCC->iStatusId = ERR_SERVICE_RSLT;
FormatMenu(pOut,pTPCC);
return(TRUE);
};
};
if (pnod->iTPRslt == SVC_BADITEMID)
pTPCC->iStatusId = INVALID_IID;

iInx = FormatRespHdr(pOut,"TPC-C New Order",pTPCC);
if (!pnod->bTPRslt)
{
iInx += wsprintf(pOut + iInx,

```

```

"<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: %-16s Credit: %-2s ",
pnod->w_id,pnod->d_id,
pnod->o_entry_d.day,pnod->o_entry_d.month,
pnod->o_entry_d.year,pnod->o_entry_d.hour,
pnod->o_entry_d.minute,pnod->o_entry_d.second,
pnod->c_id,pnod->c_last,pnod->c_credit);
iInx += sprintf(pOut + iInx,
"%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>",
pnod->c_discount * 100,pnod->o_id,pnod->o_ol_cnt,pnod->w_tax *
100,pnod->d_tax * 100);
for (u = 0; u < (UINT) pnod->o_ol_cnt; u++)
{
iInx += sprintf(pOut + iInx,
"%4.4d %6.6d %-24s %2.2d %3.3d %1.1s %6.2f
$%7.2f <BR>",
pnod->Ol[u].ol_supply_w_id,pnod->Ol[u].ol_i_id,
pnod->Ol[u].ol_i_name,pnod->Ol[u].ol_quantity,pnod-
>Ol[u].ol_stock,
pnod->Ol[u].ol_brand_generic,pnod->Ol[u].ol_i_price,
pnod->Ol[u].ol_amount);
};
for(; u < MAX_OL; u++)
{
strcat(pOut + iInx," <BR>");
iInx += 5;
};
sprintf(pOut + iInx,
"Execution Status: %24.24s Total: $%8.2f "
"</PRE><HR><BR>%s</FORM>%s",
pnod->execution_status,pnod->total_amount,
szMenuList,HTMLTrailer);
} // !bTPRslt
else
{
iInx += wsprintf(pOut + iInx,
"<PRE> New Order<BR>"
"Warehouse: %4.4d District: %2.2d
Date:<BR>"
"Customer: %4.4d Name: %-16s Credit: %-2s "
"%Disc:<BR>"
"Order Number: %8.8d Number of Lines: W_tax:
D_tax:<BR><BR>"
" Supp_W Item_Id Item Name Qty Stock B/G
Price Amount<BR>",
pnod->w_id,pnod->d_id,
pnod->c_id,pnod->c_last,pnod->c_credit,
pnod->o_id);
for(u = 0; u < MAX_OL; u++)
{
strcat(pOut + iInx," <BR>");
iInx += 5;
};
iInx += wsprintf(pOut + iInx,
"Execution Status: %24.24s Total:"

```

```

        "</PRE><HR><BR>%s</FORM>%s",
        pnod->execution_status,szMenuList,HTMLTrailer);
}; // bTPRslt

return(FALSE);

}; // ProcessNewOrder

//=====
//
// Function name: ProcessPayment
//
// ProcessPayment extracts the input data fields from pIn, processes
// the data, and returns a response in pOut.
//
// Result:
// FALSE - Payment processed successfully.
// TRUE - Payment processing failed.
//=====
BOOL ProcessPayment(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC)
{
    PAYMENT_DATA * ppd;
    TMON_STATE * pTMon;
    CHAR szWork1[60];
    CHAR szZip1[20];
    CHAR szZip2[20];
    CHAR szZip3[20];
    INT iInx;
    HRESULT hr;
    int iSize;

    pTMon = &pTPCC->tsTMon;
    ppd = (PAYMENT_DATA *) pTMon->pTxnData;
    ZeroMemory(ppd,sizeof(PAYMENT_DATA));
    ppd->w_id = pTPCC->sWid;
    // Get and validate DID
    if (GetShortKey(&ppd->d_id,pIn,"DID*",pTPCC))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    if (ppd->d_id < MIN_DID || ppd->d_id > MAX_DID)
    {
        wsprintf(pTPCC->ErrTxt,"DID Out of Range(%ld,%ld) - %ld",
            MIN_DID,MAX_DID,ppd->d_id);
        pTPCC->iStatusId = ERR_DID_INVALID;
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    // Get and validate customer Id and name
    if (GetLongKey(&ppd->c_id,pIn,"CID*",pTPCC))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    if (GetStringKey(ppd->c_last,pIn,"CLT*",pTPCC,NAME_LEN))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
};

```

```

if (ppd->c_id == 0 && ppd->c_last[0] == 0)
{
    strcpy(pTPCC->ErrTxt,"Error - Customer Id and Name Empty");
    pTPCC->iStatusId = ERR_IDANDNAME_EMPTY;
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
if (ppd->c_id != 0 && ppd->c_last[0] != 0)
{
    strcpy(pTPCC->ErrTxt,
        "Error - Specify Customer Id or Name, not Both");
    pTPCC->iStatusId = ERR_IDANDNAME_ENTERED;
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
// Get and validate customer DID
if (GetShortKey(&ppd->c_d_id,pIn,"CDI*",pTPCC))
{
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
if (ppd->c_d_id < MIN_DID || ppd->c_d_id > MAX_DID)
{
    wsprintf(pTPCC->ErrTxt,"Cust Did Out of Range(%ld,%ld) - %ld",
        MIN_DID,MAX_DID,ppd->d_id);
    pTPCC->iStatusId = ERR_DID_INVALID;
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
// Get and validate customer WID
if (GetShortKey(&ppd->c_w_id,pIn,"CWI*",pTPCC))
{
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
if (ppd->c_w_id < 1)
{
    wsprintf(pTPCC->ErrTxt,
        "Payment Contains Invalid Customer WId %d",
        ppd->c_w_id);
    pTPCC->iStatusId = ERR_WID_INVALID;
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
// Get and validate amount
if (GetAmountKey(&ppd->h_amount,pIn,"HAM*",pTPCC))
{
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
if (ppd->h_amount <= 0)
{
    wsprintf(pTPCC->ErrTxt,
        "Payment Amount Negative or Missing");
    pTPCC->iStatusId = ERR_AMOUNT_INVALID;
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
iSize = pTMon->iSize;
hr = pTMon->pIAllTxn->Payment(&iSize,(unsigned char*)&pTMon-
>pTxnData);

```

```

if (FAILED(hr))
{
    pTPCC->iStatusId = ERR_TM_INTERFACE;
    wsprintf(pTPCC->ErrTxt,
        "COM Interface to Payment Call Failed, HRESULT %x",
        hr);
    FormatMenu(pOut, pTPCC);
    return(TRUE);
};
ppd = (PAYMENT_DATA *) pTMon->pTxnData;
if (ppd->bTPRslt)
{
    wsprintf(pTPCC->ErrTxt,
        "Payment Service Returned Error(%ld): %s",
        ppd->iTPRslt, ppd->execution_status);
    pTPCC->iStatusId = ERR_SERVICE_RSLT;
    FormatMenu(pOut, pTPCC);
    return(TRUE);
};
iInx = FormatRespHdr(pOut, "TPC-C Payment", pTPCC);
FormatString(szZip1, ZIPPIC, ppd->w_zip);
FormatString(szZip2, ZIPPIC, ppd->d_zip);
FormatString(szZip3, ZIPPIC, ppd->c_zip);
FormatString(szWork1, "XXXXXX-XXX-XXX-XXXX", ppd->c_phone);
iInx += wsprintf(pOut + iInx,
    "<PRE>
    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: %-2s<BR>"
    "
    %-20s
    ",
    ppd->h_date.day, ppd->h_date.month,
    ppd->h_date.year, ppd->h_date.hour,
    ppd->h_date.minute, ppd->h_date.second,
    ppd->w_id, ppd->d_id,
    ppd->w_street_1, ppd->d_street_1,
    ppd->w_street_2, ppd->d_street_2,
    ppd->w_city, ppd->w_state, szZip1, ppd->d_city, ppd->d_state, szZip2,
    ppd->c_id, ppd->c_w_id, ppd->c_d_id,
    ppd->c_first, ppd->c_middle, ppd->c_last,
    ppd->c_since.day, ppd->c_since.month, ppd->c_since.year,
    ppd->c_street_1, ppd->c_credit, ppd->c_street_2);
iInx += sprintf(pOut + iInx, "%Disc: %5.2f<BR>", ppd->c_discount *
100);
iInx += wsprintf(pOut + iInx,
    "
    %-20s %-2s %10.10s Phone: %-19.19s<BR><BR>",
    ppd->c_city, ppd->c_state, szZip3, szWork1);
iInx += sprintf(pOut + iInx,
    "Amount Paid: $%7.2f New Cust Balance: $%14.2f<BR>"
    "Credit Limit: $%13.2f<BR><BR>",
    ppd->h_amount, ppd->c_balance, ppd->c_credit_lim);
if (ppd->c_credit[0] == 'B' && ppd->c_credit[1] == 'C')
{
    wsprintf(pOut + iInx,
        "Cust-Data: %-50.50s<BR> %-50.50s<BR>
        "%-50.50s<BR> %-50.50s<BR>"

```

```

    "</PRE><HR><BR>%s</FORM>%s",
    ppd->c_data, (ppd->c_data + 50), (ppd->c_data + 100), (ppd->c_data +
150),
    szMenuList, HTMLTrailer);
}
else
{
    wsprintf(pOut + iInx,
        "Cust-Data: <BR><BR><BR><BR>"
        "</PRE><HR><BR>%s</FORM>%s",
        szMenuList, HTMLTrailer);
};
return(FALSE);
}; // ProcessPayment
//=====
//
// Function name: ProcessDelivery
//
// ProcessDelivery extracts the input data fields from pIn, processes
// the data, and returns a response in pOut.
//
// Result:
// FALSE - Delivery processed successfully.
// TRUE - Delivery processing failed.
//=====
BOOL ProcessDelivery(CHAR * pIn, CHAR * pOut, TPCC_STATE * pTPCC)
{
    DELIVERY_DATA * pdd;
    TMON_STATE * pTMon;
    INT iInx;

    pTMon = &pTPCC->tsTMon;
    pdd = (DELIVERY_DATA *) pTMon->pTxnData;
    ZeroMemory(pdd, sizeof(DELIVERY_DATA));
    pdd->w_id = pTPCC->swid;
    // Get and validate carrier id
    if (GetShortKey(&pdd->o_carrier_id, pIn, "OCD*", pTPCC))
    {
        FormatMenu(pOut, pTPCC);
        return(TRUE);
    };
    if (pdd->o_carrier_id < MIN_CARRIER ||
        pdd->o_carrier_id > MAX_CARRIER)
    {
        wsprintf(pTPCC->ErrTxt, "Carrier Id Out of Range(%ld,%ld) - %ld",
            MIN_CARRIER, MAX_CARRIER, pdd->o_carrier_id);
        pTPCC->iStatusId = ERR_CARRIER_INVALID;
        FormatMenu(pOut, pTPCC);
        return(TRUE);
    };
    GetLocalTime(&pdd->QTime);
    DeliveryPost(pdd);
    if (pdd->bTPRslt)
    {
        wsprintf(pTPCC->ErrTxt,
            "Delivery Post Returned Error(%ld): Queue Request Failed",
            pdd->iTPRslt);

```

```

    pTPCC->iStatusId = ERR_SERVICE_RSLT;
    FormatMenu(pOut,pTPCC);
    return(TRUE);
};
iInx = FormatRespHdr(pOut,"TPC-C Delivery",pTPCC);
iInx += sprintf(pOut + iInx,
    "<PRE>
    Warehouse: %4.4d<BR><BR>"
    "Carrier Number: %2.2d<BR><BR>"
    "Execution Status: %25.25s<BR>"
    "</PRE><HR><BR>%s</FORM>%s",
    pdd->w_id,pdd->o_carrier_id,"Delivery has been queued.",
    szMenuList,HTMLTrailer);

return(FALSE);
}; // ProcessDelivery

//=====
//
// Function name: ProcessOrderStatus
//
// ProcessOrderStatus extracts the input data fields from pIn,
// processes the data, and returns a response in pOut.
//
// Result:
// FALSE - OrderStatus processed successfully.
// TRUE - OrderStatus processing failed.
//
//=====
BOOL ProcessOrderStatus(CHAR * pIn,CHAR * pOut,TPCC_STATE * pTPCC)
{
    ORDER_STATUS_DATA * posd;
    TMON_STATE * pTMon;
    INT i;
    INT iInx;
    HRESULT hr;
    int iSize;

    pTMon = &pTPCC->tsTMon;
    posd = (ORDER_STATUS_DATA *) pTMon->pTxnData;
    ZeroMemory(posd,sizeof(ORDER_STATUS_DATA));
    posd->w_id = pTPCC->sWid;
    if (GetShortKey(&posd->d_id,pIn,"DID*",pTPCC))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    if (posd->d_id < MIN_DID || posd->d_id > MAX_DID)
    {
        sprintf(pTPCC->ErrTxt,"DID Out of Range(%ld,%ld) - %ld",
            MIN_DID,MAX_DID,posd->d_id);
        pTPCC->iStatusId = ERR_DID_INVALID;
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    if (GetLongKey(&posd->c_id,pIn,"CID*",pTPCC))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
};

```

```

    if (GetStringKey(posd->c_last,pIn,"CLT*",pTPCC,NAME_LEN))
    {
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    if (posd->c_id == 0 && posd->c_last[0] == 0)
    {
        strcpy(pTPCC->ErrTxt,"Error - Customer Id and Name Empty");
        pTPCC->iStatusId = ERR_IDANDNAME_EMPTY;
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    if (posd->c_id != 0 && posd->c_last[0] != 0)
    {
        strcpy(pTPCC->ErrTxt,
            "Error - Specify Customer Id or Name, not Both");
        pTPCC->iStatusId = ERR_IDANDNAME_ENTERED;
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    iSize = pTMon->iSize;
    hr = pTMon->pIAllTxn->OrderStatus(&iSize,(unsigned char*)&pTMon-
    >pTxnData);
    if (FAILED(hr))
    {
        pTPCC->iStatusId = ERR_TM_INTERFACE;
        sprintf(pTPCC->ErrTxt,
            "COM Interface to OrderStatus Call Failed, HRESULT %x",
            hr);
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    posd = (ORDER_STATUS_DATA *) pTMon->pTxnData;
    if (posd->bTPRslt)
    {
        sprintf(pTPCC->ErrTxt,
            "Order Status Service Returned Error(%ld): %s",
            posd->iTPRslt,posd->execution_status);
        pTPCC->iStatusId = ERR_SERVICE_RSLT;
        FormatMenu(pOut,pTPCC);
        return(TRUE);
    };
    iInx = FormatRespHdr(pOut,"TPC-C Order-Status",pTPCC);
    iInx += sprintf(pOut + iInx,
        "<PRE>
        Order-Status<BR>"
        "Warehouse: %4.4d District: %2.2d<BR>"
        "Customer: %4.4d Name: %-16s %-2s %-16s<BR>",
        posd->w_id,posd->d_id,
        posd->c_id,posd->c_first,posd->c_middle,posd->c_last);
    iInx += sprintf(pOut + iInx,"Cust-Balance: $%9.2f<BR><BR>",posd-
    >c_balance);
    iInx += sprintf(pOut + iInx,
        "Order-Number: %8.8d Entry-Date: %2.2d-%2.2d-%4.4d
    %2.2d:%2.2d:%2.2d Carrier-Number: %2.2d<BR>"
        "Supply-W Item-Id Qty Amount Delivery-Date<BR>",
        posd->o_id,posd->o_entry_d.day,posd->o_entry_d.month,
        posd->o_entry_d.year,posd->o_entry_d.hour,
        posd->o_entry_d.minute,posd->o_entry_d.second,
        posd->o_carrier_id);
    for(i = 0; i < posd->o_ol_cnt; i++)
    {

```

```

        iInx += sprintf(pOut + iInx,
            " %4.4d %6.6d %2.2d %8.2f %2.2d-%2.2d-%4.4d<BR>",
            posd->OlOrderStatusData[i].ol_supply_w_id,
            posd->OlOrderStatusData[i].ol_i_id,
            posd->OlOrderStatusData[i].ol_quantity,
            posd->OlOrderStatusData[i].ol_amount,
            posd->OlOrderStatusData[i].ol_delivery_d.day,
            posd->OlOrderStatusData[i].ol_delivery_d.month,
            posd->OlOrderStatusData[i].ol_delivery_d.year);
    };
    wsprintf(pOut + iInx,
        "<BR></PRE><HR><BR>%s</FORM>%s", szMenuList, HTMLTrailer);

    return(FALSE);
}; // ProcessOrderStatus

//=====
//
// Function name: ProcessStockLevel
//
// ProcessStockLevel extracts the input data fields from pIn,
// processes the data, and returns a response in pOut.
//
// Result:
// FALSE - StockLevel processed successfully.
// TRUE - StockLevel processing failed.
//
//=====
BOOL ProcessStockLevel(CHAR * pIn, CHAR * pOut, TPCC_STATE * pTPCC)
{
    STOCK_LEVEL_DATA * psld;
    TMON_STATE * pTMon;
    HRESULT hr;
    INT iInx;
    int iSize;

    pTMon = &pTPCC->tsTMon;
    psld = (STOCK_LEVEL_DATA *) pTMon->pTxnData;
    ZeroMemory(psld, sizeof(STOCK_LEVEL_DATA));
    psld->w_id = pTPCC->sWid;
    psld->d_id = pTPCC->sDId;
    psld->low_stock = 0;
    psld->execution_status[0] = 0;
    if (GetShortKey(&psld->thresh_hold, pIn, "TT*", pTPCC))
    {
        FormatMenu(pOut, pTPCC);
        return(TRUE);
    };
    if (psld->thresh_hold < MIN_THRESHOLD ||
        psld->thresh_hold > MAX_THRESHOLD)
    {
        wsprintf(pTPCC->ErrTxt, "Threshold Out of Range(%ld,%ld) - %ld",
            MIN_THRESHOLD, MAX_THRESHOLD, psld->thresh_hold);
        pTPCC->iStatusId = ERR_THRESHOLD_RANGE;
        FormatMenu(pOut, pTPCC);
        return(TRUE);
    };
    iSize = pTMon->iSize;

```

```

    hr = pTMon->pIAllTxn->StockLevel(&iSize, (unsigned char**) &pTMon->pTxnData);
    if (FAILED(hr))
    {
        pTPCC->iStatusId = ERR_TM_INTERFACE;
        wsprintf(pTPCC->ErrTxt,
            "COM Interface to StockLevel Call Failed, HRESULT %x",
            hr);
        FormatMenu(pOut, pTPCC);
        return(TRUE);
    };
    psld = (STOCK_LEVEL_DATA *) pTMon->pTxnData;
    if (psld->bTPRslt)
    {
        wsprintf(pTPCC->ErrTxt,
            "Stock Level Service Returned Error(%ld): %s",
            psld->iTPRslt, psld->execution_status);
        pTPCC->iStatusId = ERR_SERVICE_RSLT;
        FormatMenu(pOut, pTPCC);
        return(TRUE);
    };
    iInx = FormatRespHdr(pOut, "TPC-C Stock Level", pTPCC);
    wsprintf(pOut + iInx,
        "<PRE>
        Warehouse: %4.4d District: %2.2d<BR><BR>"
        "Stock Level Threshold: %2.2d<BR><BR>"
        "low stock: %3.3ld</PRE><BR><HR>"
        "%s</FORM>%s",
        pTPCC->sWid, pTPCC->sDId, psld->thresh_hold, psld->low_stock,
        szMenuList, HTMLTrailer);

    return(FALSE);
}; // ProcessStockLevel

//=====
//
// Function name: GetHidden
//
//=====
BOOL GetHidden(CHAR * pMsg, UINT * uFormId, INT * iSyncId, INT * iTermId)
{
    CHAR * pPtr;
    BOOL bRslt = TRUE;

    // Extract TERMID
    pPtr = strstr(pMsg, TERMIDTOKEN);
    if (pPtr == NULL)
        goto xit;
    pPtr += strlen(TERMIDTOKEN);
    *iTermId = atoi(pPtr);

    // Extract SYNCID
    pPtr = strstr(pMsg, SYNCIDTOKEN);
    if (pPtr == NULL)
        goto xit;
    pPtr += strlen(SYNCIDTOKEN);
    *iSyncId = atoi(pPtr);

    // Extract FORMID
    pPtr = strstr(pMsg, FORMIDTOKEN);

```

```

if (pPtr == NULL)
    goto xit;
pPtr += strlen(FORMIDTOKEN);
*uFormId = abs(atoi(pPtr));

bRslt = FALSE;

xit:

    return(bRslt);
}; // GetHidden

//=====
//
// Function name: GetCmd
//
//=====
BOOL GetCmd(CHAR * pMsg,CHAR * pWork,UINT uLen)
{
    UINT u;
    CHAR * ptr;
    CHAR * pUpd;

    // Check for CMD key
    if (!(ptr = strstr(pMsg,CMDTOKEN)))
        return(CMD_NULL);
    ptr += sizeof(CMDTOKEN);
    pUpd = pWork;
    while (*ptr && *ptr != '&')
        *pUpd++ = *ptr++;
    *pUpd = 0;

    // Convert command name into command index
    for(u=0; u < CMD_MAX; u++)
    {
        if (!strcmp(szCmds[u],pWork))
            return(u);
    };

    // Command string not found
    return(CMD_NULL);
}; // GetCmd

//=====
//
// Function name: GetLongKey
//
//=====
BOOL GetLongKey(LONG * lRslt,CHAR * pHTML,CHAR * pKey,TPCC_STATE * pTPCC)
{
    if (GetKeyValue(pHTML,pKey,pTPCC->szWork,sizeof(pTPCC->szWork)))
    {
        wsprintf(pTPCC->ErrTxt,"Error - Missing %s Key",pKey);
        pTPCC->iStatusId = ERR_MISSING_KEY;
        return(TRUE);
    };
    if (pTPCC->szWork[0] != 0 )
    {
        if (CheckNumeric(pTPCC->szWork))
        {
            wsprintf(pTPCC->ErrTxt,"Error - %s Value Not Numeric",pKey);
            pTPCC->iStatusId = ERR_NOT_NUMERIC;
            return(TRUE);
        };
        *lRslt = atoi(pTPCC->szWork);
        return(FALSE);
    };
}; // GetLongKey

```

```

{
    wsprintf(pTPCC->ErrTxt,"Error - %s Value Not Numeric",pKey);
    pTPCC->iStatusId = ERR_NOT_NUMERIC;
    return(TRUE);
};
};
*lRslt = atoi(pTPCC->szWork);
return(FALSE);
}; // GetLongKey

//=====
//
// Function name: GetIntKey
//
//=====
BOOL GetIntKey(INT * iRslt,CHAR * pHTML,CHAR * pKey,TPCC_STATE * pTPCC)
{
    if (GetKeyValue(pHTML,pKey,pTPCC->szWork,sizeof(pTPCC->szWork)))
    {
        wsprintf(pTPCC->ErrTxt,"Error - Missing %s Key",pKey);
        pTPCC->iStatusId = ERR_MISSING_KEY;
        return(TRUE);
    };
    if (pTPCC->szWork[0] != 0 )
    {
        if (CheckNumeric(pTPCC->szWork))
        {
            wsprintf(pTPCC->ErrTxt,"Error - %s Value Not Numeric",pKey);
            pTPCC->iStatusId = ERR_NOT_NUMERIC;
            return(TRUE);
        };
    };
    *iRslt = atoi(pTPCC->szWork);
    return(FALSE);
}; // GetIntKey

//=====
//
// Function name: GetShortKey
//
//=====
BOOL GetShortKey(SHORT * sRslt,CHAR * pHTML,CHAR * pKey,TPCC_STATE * pTPCC)
{
    if (GetKeyValue(pHTML,pKey,pTPCC->szWork,sizeof(pTPCC->szWork)))
    {
        wsprintf(pTPCC->ErrTxt,"Error - Missing %s Key",pKey);
        pTPCC->iStatusId = ERR_MISSING_KEY;
        return(TRUE);
    };
    if (pTPCC->szWork[0] != 0 )
    {
        if (CheckNumeric(pTPCC->szWork))
        {
            wsprintf(pTPCC->ErrTxt,"Error - %s Value Not Numeric",pKey);
            pTPCC->iStatusId = ERR_NOT_NUMERIC;
            return(TRUE);
        };
    };
    *sRslt = (SHORT) atoi(pTPCC->szWork);
    return(FALSE);
};

```

```

}; // GetShortKey

//=====
//
// Function name: GetStringKey
//
//=====
BOOL GetStringKey(CHAR * szRslt,CHAR * pHTML,CHAR * pKey,
                 TPCC_STATE * pTPCC,UINT uMax)
{
    UINT uLen;
    if (GetKeyValue(pHTML,pKey,pTPCC->szWork,sizeof(pTPCC->szWork)))
    {
        wsprintf(pTPCC->ErrTxt,"Error - Missing %s Key",pKey);
        pTPCC->iStatusId = ERR_MISSING_KEY;
        return(TRUE);
    };
    uLen = strlen(pTPCC->szWork);
    if (uLen > uMax)
    {
        wsprintf(pTPCC->ErrTxt,
            "Error - %s Key Input (%ld) Too Long (%ld)"
            ,pKey,uLen,uMax);
        pTPCC->iStatusId = ERR_INPUT_TOOLONG;
        return(TRUE);
    };
    _strupr(pTPCC->szWork);
    strcpy(szRslt,pTPCC->szWork);
    return(FALSE);
}; // GetStringKey

//=====
//
// Function name: GetAmountKey
//
//=====
BOOL GetAmountKey(DOUBLE * dRslt,CHAR * pHTML,CHAR * pKey,
                 TPCC_STATE * pTPCC)
{
    CHAR * ptr;
    BOOL bInvalid = FALSE;

    if (GetKeyValue(pHTML,pKey,pTPCC->szWork,sizeof(pTPCC->szWork)))
    {
        wsprintf(pTPCC->ErrTxt,"Error - Missing %s Key",pKey);
        pTPCC->iStatusId = ERR_MISSING_KEY;
        return(TRUE);
    };
    ptr = pTPCC->szWork;
    while(*ptr)
    {
        if (*ptr == '.')
        {
            ptr++;
            if (!*ptr)
                break;
            if (*ptr < '0' || *ptr > '9')
            {
                bInvalid = TRUE;
                break;
            };
        };
    };
};

```

```

ptr++;
if (!*ptr)
    break;
if (*ptr < '0' || *ptr > '9')
{
    bInvalid = TRUE;
    break;
};
ptr++;
if (*ptr)
{
    bInvalid = TRUE;
    break;
};
break;
}
else
if (*ptr < '0' || *ptr > '9')
{
    bInvalid = TRUE;
    break;
};
ptr++;
}; // while(!*ptr)

if (!bInvalid)
*dRslt = atof(pTPCC->szWork);
else
{
    wsprintf(pTPCC->ErrTxt,
        "Error - Invalid Amount Format (%s)",pTPCC->szWork);
    pTPCC->iStatusId = ERR_AMOUNT_BADFORM;
};

return(bInvalid);
}; // GetAmountKey

//=====
//
// Function name: GetKeyValue
// This function parses an HTTP formatted string for specific key
// values. HTTP keys terminate with '='. HTTP values terminate
// with an '&' or '\0'.
//
// Result:
// FALSE - Key found, string value return in pValue
// TRUE - Key not found
//
//=====
BOOL GetKeyValue(CHAR * pHTML,CHAR * pKey,CHAR * pValue,UINT uMax)
{
    CHAR * ptr;
    if (!(ptr=strstr(pHTML,pKey)))
        return(TRUE);
    if (!(ptr=strchr(ptr,'='))
        return(TRUE);
    ptr++;
    uMax--;
    while (*ptr && *ptr != '&' && uMax)
    {
};

```



```

    *pValue++ = *ptr++;
    uMax--;
};
*pValue = 0;
return(FALSE);
}; // GetKeyValue

//=====
//
// Function name: FormatLogin
//
//=====
VOID FormatLogin(CHAR * pOut,CHAR * pAddText)
{
    wsprintf(pOut,"%s<BR>%s<BR>%s",szFormLogin,pAddText,HTMLTrailer);
}; // FormatLogin

//=====
//
// Function name: FormatMenu
//
//=====
VOID FormatMenu(CHAR * pOut,TPCC_STATE * pTPCC)
{
    wsprintf(pOut,
        "%s<HTML><HEAD><TITLE>TPC-C MainMenu</TITLE></HEAD><BODY>"
        "Select Desired Transaction.<BR><HR>"
        "<FORM ACTION=\"tpcc.dll\"METHOD=\"GET\">"
        "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMINID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">"
        "%s</FORM><BR>%s<BR>%s",
        HTTPHdr,pTPCC->iStatusId,pTPCC->iTermId,pTPCC->iSyncId,FORM_MENU,
        szMenuList,pTPCC->ErrTxt,HTMLTrailer);
}; // FormatMenu

//=====
//
// Function name: FormatNewOrder
//
//=====
VOID FormatNewOrder(CHAR * pOut,TPCC_STATE * pTPCC)
{
    INT iInx;
    pTPCC->uFormId = FORM_NEWORDER;
    iInx = FormatFormHdr(pOut,"TPC-C New Order",pTPCC);
    iInx += wsprintf(pOut + iInx,
        "<PRE>
        "Warehouse: %4.4d District: <INPUT NAME=\"DID*\" SIZE=1>
Date:<BR>",
        pTPCC->sWId);
        strcpy(pOut + iInx,
        "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>"
        " <INPUT NAME=\"SP00*\" SIZE=4> <INPUT NAME=\"IID00*\" SIZE=6>
<INPUT NAME=\"Qty00*\" 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></BODY></HTML>");
}; // FormatNewOrder

//=====
//
// Function name: FormatPayment
//
//=====
VOID FormatPayment(CHAR * pOut,TPCC_STATE * pTPCC)
{
    INT iInx;
    pTPCC->uFormId = FORM_PAYMENT;
    iInx = FormatFormHdr(pOut,"TPC-C Payment",pTPCC);
    iInx += wsprintf(pOut + iInx,
        "<PRE>
        "Date:<BR><BR>"
        "Warehouse: %4.4d",
        pTPCC->sWId);
        strcpy(pOut + iInx,
        " 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\">"
        "</FORM></BODY></HTML>");
}; // FormatPayment

//=====
//
// Function name: FormatDelivery
//
//=====
VOID FormatDelivery(CHAR * pOut,TPCC_STATE * pTPCC)
{
    INT iInx;
    pTPCC->uFormId = FORM_DELIVERY;
    iInx = FormatFormHdr(pOut,"TPC-C Delivery",pTPCC);
    wsprintf(pOut + iInx,
        "<PRE>                                Delivery<BR>"
        "Warehouse: %4.4d<BR><BR>"
        "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\">"
        "</FORM></BODY></HTML>",
        pTPCC->sWid);
}; // FormatDelivery

//=====
//
// Function name: FormatOrderStatus
//
//=====
VOID FormatOrderStatus(CHAR * pOut,TPCC_STATE * pTPCC)
{
    INT iInx;
    pTPCC->uFormId = FORM_ORDERSTATUS;
    iInx = FormatFormHdr(pOut,"TPC-C Order-Status",pTPCC);
    wsprintf(pOut + iInx,
        "<PRE>                                Order-Status<BR>"
        "Warehouse: %4.4d "
        "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\">"
        "</FORM></BODY></HTML>",
        pTPCC->sWid);
}; // FormatOrderStatus

//=====
//
// Function name: FormatStockLevel
//
//=====

```

```

VOID FormatStockLevel(CHAR * pOut,TPCC_STATE * pTPCC)
{
    INT iInx;
    pTPCC->uFormId = FORM_STOCKLEVEL;
    iInx = FormatFormHdr(pOut,"TPC-C Stock Level",pTPCC);
    wsprintf(pOut + iInx,
        "<PRE>                                Stock-Level<BR>"
        "Warehouse: %4.4d   District: %2.2d<BR><BR>"
        "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></BODY></HTML>",
        pTPCC->sWid,pTPCC->sDId);
}; // FormatStockLevel

//=====
//
// Function name: FormatFormHdr
//
//=====
INT FormatFormHdr(CHAR * pOut,CHAR * pTitle,TPCC_STATE * pTPCC)
{
    return(wsprintf(pOut,
        "%s<HTML><HEAD><TITLE>%s</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=\"FORMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMIN\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
        HTTPHdr,pTitle,pTPCC->uFormId,pTPCC->iTermId,pTPCC->iSyncId
    ));
}; // FormatFormHdr

//=====
//
// Function name: FormatRespHdr
//
//=====
INT FormatRespHdr(CHAR * pOut,CHAR * pTitle,TPCC_STATE * pTPCC)
{
    return(wsprintf(pOut,
        "%s<HTML><HEAD><TITLE>%s</TITLE></HEAD>"
        "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
        "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMIN\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">",
        HTTPHdr,pTitle,pTPCC->iStatusId,pTPCC->uFormId,
        pTPCC->iTermId,pTPCC->iSyncId
    ));
}; // FormatRespHdr

//=====
//
// Function name: FormatString
//
// Encodes formatted string for HTML transmission.
//
//=====

```

```

VOID FormatString(CHAR * pOut,CHAR * pPic,CHAR * pIn)
{
    while(*pPic)
    {
        if (*pPic == 'X' )
        {
            if (*pIn)
                *pOut++ = *pIn++;
            else
                *pOut++ = ' ';
        }
        else
            *pOut++ = *pPic;
        pPic++;
    };
    *pOut = 0;
}; // FormatString

//=====
// FUNCTION: UtilStrCpy
//
// Copies n characters from string pSrc to pDst and places a null
// null character at the end of the destination string. Unlike
// strncpy this function ensures that the result string is always
// null terminated.
//
//=====
VOID UtilStrCpy(CHAR * pDest,CHAR * pSrc,INT n)
{
    strncpy(pDest,pSrc,n);
    pDest[n] = '\0';
    return;
}; // UtilStrCpy

//=====
//
// Function name: CheckNumeric
//
// Result
// FALSE - string is all numeric
// TRUE - sting contains non-numeric characters
//
//=====
BOOL CheckNumeric(CHAR * pNum)
{
    if (*pNum == 0 )
        return(TRUE);
    while (*pNum && isdigit(*pNum))
        pNum++;
    return(*pNum);
}; // CheckNumeric

// delivery.h
//
// Copyright Unisys, 1999

```

delivery.h

```

#define DEFAULTDQSIZE 2000

bool DeliveryInit(long lSetThreads,long lSetQSize,char * pszPath);
void DeliveryTerm(void);
bool DeliveryPost(DELIVERY_DATA * pdd);

//=====
//
// Function name: DeliveryInit
//
//=====
bool DeliveryInit(long lSetThreads,long lSetQSize,char * pszPath)
{
    char szDiag[MAX_DIAG_SZ];

    lDeliveryThreads = lSetThreads;
    lDQSize = lSetQSize;
    if (lDQSize <= 0)

```

delivery.cpp

```

// delivery.cpp
//
// Copyright Unisys, 1999

#include <windows.h>
#include <stdio.h>
#include <time.h>
#include <sys\timeb.h>
#include <process.h>

#include "..\tpccsvr\tpcc.h"
#include "tmon.h"
#include "diagio.h"
#include "delivery.h"

CRITICAL_SECTION csDQRead;
CRITICAL_SECTION csDQWrite;
HANDLE hDQRead;
HANDLE hDQStart;
bool bDQStarted = FALSE;
bool bDQQuit = FALSE;
long lDeliveryThreads;
long lDQSize;
char szPath[200];
long lDeliveryActive = 0;
typedef struct
{
    bool bInUse;
    DELIVERY_DATA ddEntry;
} DELIVERY_QUEUE;
DELIVERY_QUEUE * pDQ;
long lDQNextWrite = 0;
long lDQNextRead = 0;

bool DoDQStart(void);
UINT WINAPI DoDelivery(void * Unused);
void CalculateElapsed(int * pElapsed,LPSYSTEMTIME lpBegin,
                    LPSYSTEMTIME lpEnd);

//=====
//
// Function name: DeliveryInit
//
//=====
bool DeliveryInit(long lSetThreads,long lSetQSize,char * pszPath)
{
    char szDiag[MAX_DIAG_SZ];

    lDeliveryThreads = lSetThreads;
    lDQSize = lSetQSize;
    if (lDQSize <= 0)

```

```

    lDQSize = DEFAULTDQSIZE;
    strcpy(szPath,pszPath);
    InitializeCriticalSection(&csDQRead);
    InitializeCriticalSection(&csDQWrite);
    hDQRead = CreateEvent(NULL,TRUE,FALSE,NULL);
    if (!hDQRead)
    {
        wsprintf(szDiag,"DeliveryInit: Create DQRead Event Failure (%ld)\n",
            GetLastError());
        DiagIoWrite(szDiag,DIAG_ERROR);
        return(TRUE);
    };
    pDQ = (DELIVERY_QUEUE *) calloc(lDQSize,sizeof(DELIVERY_QUEUE));
    if (pDQ == NULL)
    {
        DiagIoWrite("DeliveryInit: Allocate Delivery Queue
Failure\n",DIAG_ERROR);
        return(TRUE);
    };
    wsprintf(szDiag,
        "DeliveryInit: Threads = %ld, DQSize(entries) = %ld\n",
        lDeliveryThreads,lDQSize);
    DiagIoWrite(szDiag,DIAG_FORCE);
    return(FALSE);
}; // DeliveryInit

//=====
//
// Function name: DoDQStart
//
//=====
bool DoDQStart(void)
{
    UINT uThread;
    ULONG hThread;
    DWORD dwRslt;
    char szDiag[MAX_DIAG_SZ];
    void * Unused = NULL;
    int i;

    bDQStarted = TRUE;
    hDQStart = CreateEvent(NULL,TRUE,FALSE,NULL);
    if (!hDQStart)
    {
        wsprintf(szDiag,"DoDQStart: Create Event Failure (%ld)\n",
            GetLastError());
        DiagIoWrite(szDiag,DIAG_ERROR);
        return(TRUE);
    };
    for (i = 0; i < lDeliveryThreads; i++)
    {
        hThread =
            _beginthreadex(NULL,
                0,
                DoDelivery,
                Unused,
                0,
                &uThread);

        if (hThread == 0)
        {
            wsprintf(szDiag,

```

```

        "DoDQStart: Begin Delivery Thread(%d) Failed(%ld)\n",
            i + 1,errno);
        DiagIoWrite(szDiag,DIAG_ERROR);
        return(TRUE);
    };
    dwRslt = WaitForSingleObject(hDQStart,60000);
    if (dwRslt == WAIT_TIMEOUT)
    {
        DiagIoWrite("DoDQStart: Wait Delivery Start Timed
Out\n",DIAG_ERROR);
        return(TRUE);
    };
    if (lDeliveryActive != (i + 1))
    {
        wsprintf(szDiag,
            "DoDQStart: Delivery Thread Initialization Failed(%ld)\n",
            i + 1);
        DiagIoWrite(szDiag,DIAG_ERROR);
        return(TRUE);
    };
    ResetEvent(hDQStart);
}; // for (lDeliveryThreads)
CloseHandle(hDQStart);
return(FALSE);
}; // DoDQStart

//=====
//
// Function name: DeliveryTerm
//
//=====
void DeliveryTerm(void)
{
    int i = 0;
    bDQQuit = TRUE;
    while (i < 12 && lDeliveryActive > 0)
    {
        SetEvent(hDQRead);
        Sleep(5000);
        i++;
    };
    if (lDeliveryActive != 0)
    {
        CHAR szDiag[MAX_DIAG_SZ];
        wsprintf(szDiag,
            "DeliveryTerm: %ld DeliveryThreads still active\n",
            lDeliveryThreads);
        DiagIoWrite(szDiag,DIAG_ERROR);
    };
    free(pDQ);
    CloseHandle(hDQRead);
    DeleteCriticalSection(&csDQWrite);
    DeleteCriticalSection(&csDQRead);
    return;
}; // DeliveryTerm

//=====
//
// Function name: DeliveryPost
//
//=====

```

```

bool DeliveryPost(DELIVERY_DATA * pPost)
{
    DELIVERY_QUEUE * pDQSlot;
    DELIVERY_DATA * pddEntry;
    if (!bdQStarted)
    {
        if (DoDQStart())
        {
            pPost->bTPRslt = TRUE;
            pPost->iTPRslt = SVCERR_DQSTART;
            return(TRUE);
        }
    };
    _try
    {
        EnterCriticalSection(&csDQWrite);
        pDQSlot = &pDQ[lDQNextWrite];
        if (pDQSlot->bInUse)
        {
            char szDiag[MAX_DIAG_SZ];
            pPost->bTPRslt = TRUE;
            pPost->iTPRslt = SVCERR_DQFULL;
            wsprintf(szDiag,
                "Delivery Post: Queue Limit (%d) Exceeded\n",
                lDQSize);
            DiagIoWrite(szDiag,DIAG_ERROR);
            return(TRUE);
        };
        pddEntry = &pDQSlot->ddEntry;
        memcpy(pddEntry,pPost,sizeof(DELIVERY_DATA));
        pDQSlot->bInUse = TRUE;
        if (lDQNextWrite == lDQNextRead)
            SetEvent(hDQRead);
        lDQNextWrite++;
        if (lDQNextWrite == lDQSize)
            lDQNextWrite = 0;
    }
    _finally
    {
        LeaveCriticalSection(&csDQWrite);
    };
    pPost->bTPRslt = FALSE;
    pPost->iTPRslt = SVC_NOERROR;
    return(FALSE);
}; // DeliveryPost

//=====
//
// Function name: DoDelivery
//
//=====
UINT WINAPI DoDelivery(void * Unused)
{
    FILE *fpLog;
    char szLogTitle[300];
    bool bFlush = FALSE;
    DELIVERY_QUEUE * pDQSlot;
    DELIVERY_DATA * pddEntry;
    DELIVERY_DATA * pdd;
    TMON_STATE tsState;
    TMON_STATE * pTMon;

```

```

HRESULT hr;
int iSize;
long lMyId;
char szTMErrTxt[500];
char szDiag[MAX_DIAG_SZ];
int iElapsed;
int iInx;
long lDQReport = 0;
bool bdQReport = FALSE;

lMyId = InterlockedIncrement(&lDeliveryActive);
pTMon = &tsState;
pTMon->pIAllTxn = NULL;
pTMon->pTxnData = NULL;
pTMon->pszErrTxt = szTMErrTxt;
if (TMInit(pTMon))
{
    wsprintf(szDiag,"DoDelivery(%ld): TMInit %s\n",lMyId,szTMErrTxt);
    DiagIoWrite(szDiag,DIAG_ERROR);
    InterlockedDecrement(&lDeliveryActive);
    SetEvent(hDQStart);
    return(1);
};
wsprintf(szLogTitle,"%sdelilog%ld",szPath,lMyId);
fpLog = fopen(szLogTitle,"ab");
if (!fpLog)
{
    wsprintf(szDiag,
        "DoDelivery(%ld): LogFile %s Open Failed (%ld)\n",
        lMyId,szLogTitle,GetLastError());
    DiagIoWrite(szDiag,DIAG_ERROR);
    InterlockedDecrement(&lDeliveryActive);
    SetEvent(hDQStart);
    return(1);
};
wsprintf(szDiag,"DoDelivery(%ld): Initialized\n",lMyId);
DiagIoWrite(szDiag,DIAG_FORCE);
SetEvent(hDQStart);

while (!bdQQuit)
{
    EnterCriticalSection(&csDQRead);
    WaitForSingleObject(hDQRead,INFINITE);
    if (bdQQuit)
    {
        LeaveCriticalSection(&csDQRead);
        break;
    };
    pDQSlot = &pDQ[lDQNextRead];
    if (!pDQSlot->bInUse)
    {
        wsprintf(szDiag,
            "DoDelivery(%ld): QSlot for Read Not InUse (%ld)\n",
            lMyId);
        DiagIoWrite(szDiag,DIAG_ERROR);
        LeaveCriticalSection(&csDQRead);
        break;
    };
    pddEntry = &pDQSlot->ddEntry;
    pdd = (DELIVERY_DATA *) pTMon->pTxnData;

```

```

memcpy(pdd,pddEntry,sizeof(DELIVERY_DATA));
EnterCriticalSection(&csDQWrite);
pDQSlot->bInUse = FALSE;
LDQNextRead++;
if (LDQNextRead == LDQSize)
{
    LDQNextRead = 0;
    bDQReport = TRUE;
    LDQReport = LDQNextWrite;
};
if (LDQNextRead == LDQNextWrite)
    ResetEvent(hDQRead);
LeaveCriticalSection(&csDQWrite);
LeaveCriticalSection(&csDQRead);
// Process delivery transaction
iSize = pTMon->iSize;
hr = pTMon->pIAllTxn->Delivery(&iSize,(unsigned char**)&pTMon-
>pTxnData);
if (FAILED(hr))
{
    wsprintf(szDiag,
        "DoDelivery(%ld): COM Interface Call Failed HRESULT %x\n",
        lMyId,hr);
    DiagIoWrite(szDiag,DIAG_ERROR);
    break;
};
pdd = (DELIVERY_DATA *) pTMon->pTxnData;
GetLocalTime(&pdd->EndTime);
iElapsed = 9999999;
if (!pdd->bTPRslt)
    CalculateElapsed(&iElapsed,&pdd->QTime,&pdd->EndTime);
iInx = wsprintf(szDiag,
"%4.4d/%2.2d/%2.2d,%2.2d:%2.2d:%2.2d:%3.3d,%2.2d:%2.2d:%2.2d:%3.3d,"
"%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d\r\n",
pdd->EndTime.wYear,pdd->EndTime.wMonth,pdd->EndTime.wDay,
pdd->QTime.wHour,pdd->QTime.wMinute,
pdd->QTime.wSecond,pdd->QTime.wMilliseconds,
pdd->EndTime.wHour,pdd->EndTime.wMinute,
pdd->EndTime.wSecond,pdd->EndTime.wMilliseconds,
iElapsed,pdd->w_id,pdd->o_carrier_id,
pdd->o_id[0],pdd->o_id[1],pdd->o_id[2],pdd->o_id[3],pdd->o_id[4],
pdd->o_id[5],pdd->o_id[6],pdd->o_id[7],pdd->o_id[8],pdd->o_id[9]
);
fwrite(szDiag,iInx,1,fpLog);
if (bDQReport)
{
    wsprintf(szDiag,"DoDelivery(%ld): DQDepth
%ld\n",lMyId,LDQReport);
    DiagIoWrite(szDiag,DIAG_FORCE);
    bDQReport = FALSE;
};
}; // while !bDQQuit

if (fpLog)
    fclose(fpLog);
TMDone(pTMon);
InterlockedDecrement(&lDeliveryActive);
wsprintf(szDiag,"DoDelivery(%ld): Shutdown\n",lMyId);
DiagIoWrite(szDiag,DIAG_FORCE);
return(0);

```

```

}; // DoDelivery
//=====
//
// Function name: CalculateElapsed (milliseconds)
//
//=====
void CalculateElapsed(int * pElapsed,LPSYSTEMTIME lpBegin,
                    LPSYSTEMTIME lpEnd)
{
    int tmBegin;
    int tmEnd;
    tmBegin = (lpBegin->wHour * 3600000) + (lpBegin->wMinute * 60000) +
        (lpBegin->wSecond * 1000) + lpBegin->wMilliseconds;
    tmEnd = (lpEnd->wHour * 3600000) + (lpEnd->wMinute * 60000) +
        (lpEnd->wSecond * 1000) + lpEnd->wMilliseconds;
    *pElapsed = tmEnd - tmBegin;
    // Check for day boundary, this will function for 24 hour period but
    // will fail over a 48 hours period.
    if (*pElapsed < 0)
        *pElapsed = *pElapsed + (24 * 60 * 60 * 1000);
    return;
}; // CalculateElapsed

```

term.h

```

// term.h
//
// Copyright Unisys, 1999

#include <sys\timeb.h>

#define TMILLI_TIMEOUT 3600000 // One hour

typedef struct
{
    BOOL bInUse; // In use flag
    INT iTermId; // TermId
    LPVOID ConnID; // Connection Id
    INT iSyncId; // Sync Id
    SHORT sWId; // TPCC WareHouse Id
    SHORT sDId; // TPCC District Id
    struct_timeb tbLastAccess; // Last activity timestamp
} TERM_STATE;

BOOL TermInit(INT iSetMaxTerm);
VOID TermTerm(VOID);
TERM_STATE * TermAlloc(VOID);
TERM_STATE * TermGet(INT iTermId);
BOOL TermFree(INT iTermId);

```

term.cpp

```

// term.cpp
//
// Copyright Unisys, 1999
//

```

```

#include <windows.h>
#include <stdio.h>
#include "diagio.h"
#include "timesupp.h"
#include "term.h"

TERM_STATE * pTArray;
INT iNextTerm = 0;
INT iMaxTerm = 0;
CRITICAL_SECTION csTerm;

VOID TermMaint(VOID);

//=====
//
// Function name: TermInit
// Creates and initializes the first TERMINITIAL TArray entries.
// Initializes critical section to control access to TArray. Assumes
// access to function is single threaded, no other threads will start
// until this function completes and that function is called once
// (DLL_PROCESS_ATTACH).
//
// Returns:
// FALSE TArray allocated and initialized
// TRUE TArray allocation failure
//=====
BOOL TermInit(INT iSetMaxTerm)
{
    INT iTermId;
    CHAR szDiag[MAX_DIAG_SZ];
    if (pTArray != NULL)
    {
        wsprintf(szDiag, "TermInit(%ld): TArray Already Initialized\n",
            GetCurrentThreadId());
        DiagIoWrite(szDiag, DIAG_ERROR);
        return(TRUE);
    };
    InitializeCriticalSection(&csTerm);
    iMaxTerm = iSetMaxTerm;
    pTArray = (TERM_STATE *) malloc(sizeof(TERM_STATE) * (iMaxTerm + 1));
    if (pTArray == NULL)
    {
        wsprintf(szDiag, "TermInit(%ld): malloc failed (%ld)\n",
            GetCurrentThreadId(), GetLastError());
        DiagIoWrite(szDiag, DIAG_ERROR);
        return(TRUE);
    }
    for (iTermId = 1; iTermId <= iMaxTerm; iTermId++)
        TermFree(iTermId);
    iNextTerm = 1;
    return(FALSE);
}; // TermInit

//=====
//
// Function name: TermTerm
// Frees TArray and deletes csTerm critical section. Assumes access
// to function is single threaded and no other threads are actively
// accessing TArray entries (DLL_PROCESS_DETACH).
//

```

```

//=====
VOID TermTerm(VOID)
{
    DeleteCriticalSection(&csTerm);
    if (pTArray != NULL)
        free(pTArray);
    iNextTerm = 0;
    iMaxTerm = 0;
}; // TermTerm

//=====
//
// Function name: TermAlloc
// Allocates empty TArray. Uses iNextTerm to start search.
//
// Returns:
// > 0 TArray entry index (iTermId)
// < 0 Empty TArray entry not available
//=====
TERM_STATE * TermAlloc(VOID)
{
    INT iTermId = -1;
    if (pTArray == NULL)
    {
        CHAR szDiag[MAX_DIAG_SZ];
        wsprintf(szDiag, "TermAlloc(%ld): Term Array Not Allocated\n",
            GetCurrentThreadId());
        DiagIoWrite(szDiag, DIAG_ERROR);
        return(NULL);
    };
    EnterCriticalSection(&csTerm);
    __try
    {
        while(iNextTerm <= iMaxTerm)
        {
            if (!pTArray[iNextTerm].bInUse)
            {
                pTArray[iNextTerm].bInUse = TRUE;
                _ftime(&pTArray[iNextTerm].tbLastAccess);
                iTermId = iNextTerm;
                iNextTerm++;
                break;
            };
            iNextTerm++;
        }; // while(iNextTerm <= iMaxTerm) (1st Try)
        if (iTermId <= 0)
        {
            // No entry found. Perform maint and try again
            TermMaint();
            iNextTerm = 1;
            while(iNextTerm <= iMaxTerm)
            {
                if (!pTArray[iNextTerm].bInUse)
                {
                    pTArray[iNextTerm].bInUse = TRUE;
                    _ftime(&pTArray[iNextTerm].tbLastAccess);
                    iTermId = iNextTerm;
                    iNextTerm++;
                    break;
                };
            };
        };
    };
}

```

```

        iNextTerm++;
    }; // while(iNextTerm <= iMaxTerm) (2nd Try)
}; // if (iTermId <= 0)
if (iTermId <= 0)
    iNextTerm = 1;
}
_finally
{
    LeaveCriticalSection(&csTerm);
};

if (iTermId > 0)
    return(&TArray[iTermId]);
else
    return(NULL);

}; // TermAlloc

//=====
//
// Function name: TermMaint
// Clears entries whose last access time exceeds TMILLI_TIMEOUT.
// Assumes caller has entered csTerm.
//
//=====
VOID TermMaint(VOID)
{
    INT iTermId;
    TMILLI tmElapsed;
    // Free entries that have timed out
    for (iTermId = 1; iTermId <= iMaxTerm; iTermId++)
    {
        if (pTArray[iTermId].bInUse)
        {
            tmElapsed = TimebElapsed(&pTArray[iTermId].tbLastAccess);
            if (tmElapsed > TMILLI_TIMEOUT)
                TermFree(iTermId);
        }
    };
}; // TermMaint

//=====
//
// Function name: TermGet
// Returns pointer to TArray slot at iTermId.
//
// Returns:
// FALSE TArray entry made available
// TRUE iTermId invalid.
//
//=====
TERM_STATE * TermGet(INT iTermId)
{
    TERM_STATE * pTerm;
    TMILLI tmElapsed;
    if (iTermId <= 0 || iTermId > iMaxTerm)
    {
        CHAR szDiag[MAX_DIAG_SZ];
        sprintf(szDiag, "TermGet(%ld): Invalid TermId (%ld)\n",
            GetCurrentThreadId(), iTermId);
    }

```

```

        DiagIoWrite(szDiag, DIAG_ERROR);
        return(NULL);
    };
    pTerm = &pTArray[iTermId];
    if (!pTerm->bInUse)
        return(NULL);
    tmElapsed = TimebElapsed(&pTerm->tbLastAccess);
    if (tmElapsed > TMILLI_TIMEOUT)
        return(NULL); // Entry destined to be freed by maint
    _ftime(&pTArray[iTermId].tbLastAccess);
    return(&pTArray[iTermId]);
}; // TermGet

//=====
//
// Function name: TermFree
// Initializes contents of TArray slot at iTermId.
//
// Returns:
// FALSE TArray entry made available
// TRUE iTermId invalid.
//
//=====
BOOL TermFree(INT iTermId)
{
    TERM_STATE * pTerm;
    if (iTermId <= 0 || iTermId > iMaxTerm)
    {
        CHAR szDiag[MAX_DIAG_SZ];
        sprintf(szDiag, "TermFree(%ld): Invalid TermId (%ld)\n",
            GetCurrentThreadId(), iTermId);
        DiagIoWrite(szDiag, DIAG_ERROR);
        return(TRUE);
    };
    pTerm = &pTArray[iTermId];
    pTerm->ConnID = 0;
    pTerm->sWid = 0;
    pTerm->sDId = 0;
    pTerm->iSyncId = 0;
    pTerm->iTermId = iTermId;
    TimebClear(&pTerm->tbLastAccess);
    pTerm->bInUse = FALSE;
    return(FALSE);
}; // TermFree

tmon.h

// tmon.h
//
// Copyright Unisys, 1999

#include "..\tpccproxy\tpccproxy.h"

typedef struct
{
    CHAR * pszErrTxt; // Error text
    INT iSize;
    ITPCC * pIAllTxn;
    CHAR * pTxnData; // TM buffer area

```



```

} TMON_STATE;

VOID TMonInit(INT iSetMaxMsg);
VOID TMonTerm(VOID);
BOOL TMinInit(TMON_STATE * pTMon);
VOID TMDone(TMON_STATE * pTMon);

```

tmon.cpp

```

// tmon.cpp
//
// Copyright Unisys, 1999
//

// needed for CoinitializeEx
#define _WIN32_WINNT 0x0400

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

#include "..\tpccproxy\tpccproxy_i.c"
#include "..\tpccsvr\tpccsvr_i.c"

#include "tmon.h"

INT iTMMMaxSz;

//=====
//
// Function name: TMonInit
//
//=====
VOID TMonInit(INT iSetMaxMsg)
{
    iTMMMaxSz = iSetMaxMsg;
}; // TMonInit

//=====
//
// Function name: TMonTerm
//
//=====
VOID TMonTerm(VOID)
{
}; // TMonTerm

//=====
//
// Function name: TMinInit
//
// Result:
// FALSE Initialization completed successfully
// TRUE Initialization failed
//
//=====
BOOL TMinInit(TMON_STATE * pTMon)
{
    HRESULT hr = NULL;

```

```

long lRet = 0;

pTMon->pIAllTxn = NULL;
pTMon->pTxnData = NULL;
pTMon->iSize = 0;
// Must have ErrTxt message area set before init
if (pTMon->pszErrTxt == NULL)
    return(TRUE);

hr = CoInitializeEx(NULL, COINIT_MULTITHREADED);
if (FAILED(hr))
{
    wsprintf(pTMon->pszErrTxt, "COM Initialize Failed, HRESULT %x\n", hr);
    return(TRUE);
};

hr = CoCreateInstance(CLSID_TPCC, NULL, CLSCTX_SERVER, IID_ITPCC,
    (void **)&pTMon->pIAllTxn);
if (FAILED(hr))
{
    wsprintf(pTMon->pszErrTxt, "COM Create Instance Failed, HRESULT
%x\n", hr);
    return(TRUE);
};
hr = (pTMon->pIAllTxn)->CallSetComplete();
if (FAILED(hr))
{
    wsprintf(pTMon->pszErrTxt, "COM Call SetComplete Failed, HRESULT
%x\n", hr);
    return(TRUE);
};
pTMon->pTxnData = (char *) CoTaskMemAlloc(iTMMMaxSz);
if (!(pTMon->pTxnData))
{
    wsprintf(pTMon->pszErrTxt, "COM Allocate TxnData Failed\n");
    return(TRUE);
};
pTMon->iSize = iTMMMaxSz;
return(FALSE);
}; // TMinInit

//=====
//
// Function name: TMDone
//
//=====
VOID TMDone(TMON_STATE * pTMon)
{
    CoTaskMemFree(pTMon->pTxnData);
    pTMon->pIAllTxn->Release();
    pTMon->pIAllTxn = NULL;
    CoUninitialize();
}; // TMDone

```

timesupp.h

```

// timesupp.h
//
// Copyright Unisys, 1999

```

```

#include <windows.h>
#include <time.h>
#include <sys\timeb.h>

#define TIMEBSEED_MOD 10000
#define TIMEBSEED_SHIFT 1000
#define TIMEB_STRING_SZ 23
#define TIMEB_STRING_DATESZ 10
#define TIMEB_STRING_TIMEOFFSET 11
#define TIMEB_STRING_TIMESZ 12

typedef ULONG TMILLI;

TMILLI TimebDiff(struct _timeb * p_t1, struct _timeb * p_t2);
VOID TimebCopy(struct _timeb * p_tDest, struct _timeb * p_tSource);
TMILLI TimebElapsed(struct _timeb * p_t1);
VOID TimebClear(struct _timeb * p_t1);
CHAR * TimebToString(struct _timeb * p_t1, CHAR * psz, BOOL bMillis);
BOOL TimebFromString(struct _timeb * p_t1, CHAR * psz);
VOID TimebAddSecs(struct _timeb * p_t1, INT iSeconds);
ULONG TimebSeed(VOID);

```

timesupp.cpp

```

// timesupp.c
//
// Copyright Unisys, 1997 1998 1999 2000
//

#include <stdio.h>
#include "timesupp.h"

//=====
//
// Function name: TimebCopy
// Structure contents copy of _timeb source to _timeb dest.
//
//=====
VOID TimebCopy(struct _timeb * p_tDest, struct _timeb * p_tSource)
{
    p_tDest->time = p_tSource->time;
    p_tDest->millitm = p_tSource->millitm;
    p_tDest->dstflag = p_tSource->dstflag;
    p_tDest->timezone = p_tSource->timezone;
}; // TimebCopy

//=====
//
// Function name: TimebDiff
// Time difference in milliseconds between _timeb_t1 and _timeb_t2.
//
//=====
TMILLI TimebDiff(struct _timeb * p_t1, struct _timeb * p_t2)
{
    LONG lRslt;
    lRslt = ((p_t2->time - p_t1->time) * 1000) +
            (p_t2->millitm - p_t1->millitm);

```

```

    if (lRslt < 0)
        return(0);
    else
        return((TMILLI) lRslt);
}; // TimebDiff

//=====
//
// Function name: TimebElapsed
//
//=====
TMILLI TimebElapsed(struct _timeb * p_t1)
{
    struct _timeb _tb2;
    _ftime(&_tb2);
    return (TimebDiff(p_t1, &_tb2));
}; // TimebElapsed

//=====
//
// Function name: TimebClear
//
//=====
VOID TimebClear(struct _timeb * p_t1)
{
    p_t1->time = 0;
    p_t1->millitm = 0;
}; // TimebClear

//=====
//
// Function name: TimebToString
// Converts timeb to yyyy:mm:dd,hh:mm:ss.sss format
//
//=====
CHAR * TimebToString(struct _timeb * p_t1, CHAR * psz, BOOL bMillis)
{
    struct tm * ptm;
    int iInx;
    ptm = localtime(&p_t1->time);
    iInx = wsprintf(psz, "%4.4d/%2.2d/%2.2d,%2.2d:%2.2d:%2.2d",
        ptm->tm_year + 1900, ptm->tm_mon + 1, ptm->tm_mday,
        ptm->tm_hour, ptm->tm_min, ptm->tm_sec);
    if (bMillis)
        wsprintf(psz + iInx, ".%3.3d", p_t1->millitm);
    return(psz);
}; // TimebToString

//=====
//
// Function name: TimebFromString
// Converts yyyy:mm:dd,hh:mm:ss.sss (TimebToString) format to timeb
//
//=====
BOOL TimebFromString(struct _timeb * p_t1, CHAR * psz)
{
    struct tm tmTime;
    struct tm * ptm;

```

```

UINT uLen;

ptm = &tmTime;
uLen = strlen(psz);
if (uLen < (TIMEB_STRING_SZ - 4)) // millis are optional
{
    p_tbl->time = 0;
    p_tbl->millitm = 0;
    return (TRUE);
};
// Clear fields that won't be set
ptm->tm_wday = 0;
ptm->tm_yday = 0;
ptm->tm_isdst = -1;
// Set tm struct fields from string
ptm->tm_year = (atoi(psz)) - 1900;
psz += 5;
ptm->tm_mon = (atoi(psz)) - 1;
psz += 3;
ptm->tm_mday = atoi(psz);
psz += 3;
ptm->tm_hour = atoi(psz);
psz += 3;
ptm->tm_min = atoi(psz);
psz +=3;
ptm->tm_sec = atoi(psz);
if (uLen >= TIMEB_STRING_SZ) // Millis present
{
    psz += 3;
    p_tbl->millitm = atoi(psz);
}
else
    p_tbl->millitm = 0;
p_tbl->time = mktime(ptm);
return(FALSE);
}; // TimebFromString

//=====
//
// Function name: TimebAddSecs
//
//=====
VOID TimebAddSecs(struct _timeb * p_tbl,INT iSeconds)
{
    p_tbl->time += iSeconds;
}; // TimebAddSecs

//=====
//
// Function name: TimebSeed
//
//=====
ULONG TimebSeed(VOID)
{
    ULONG ulSeed;
    struct _timeb tb_1;
    _ftime(&tb_1);
    ulSeed = ((tb_1.time % TIMEBSEED_MOD) * TIMEBSEED_SHIFT) +
tb_1.millitm;
    return(ulSeed);
}; // TimebSeed

```

diagio.h

```

// diagio.h
//
// Copyright Unisys, 1999

// Environment variable defaults
#define DEFAULTDIAGLEVEL DIAG_INFO
#define DEFAULTTEVENTLOG 0

#define DIAGNOSTICS TRUE
#define MAX_DIAG_SZ 2000

// Severity level of diagnostic report
#define DIAG_FORCE 1
#define DIAG_ERROR 2
#define DIAG_STATE 3
#define DIAG_INFO 4

VOID DiagIoInit(CHAR * pDiagId,BOOL bConsole,BOOL bEvent,UINT uLevel);
VOID DiagIoTerm(VOID);
VOID DiagIoWrite(CHAR * pDiagBuffer, UINT uSeverity);

```

diagio.cpp

```

// diagio.cpp
//
// Copyright Unisys, 1999
//
#include <windows.h>
#include <stdio.h>
#include "diagio.h"

CRITICAL_SECTION csDiagIo;
HANDLE hEventLog = NULL;
UINT uDiagLevel;
BOOL bEventLog;
BOOL bConsoleLog;
CHAR * pDiagHdr;
CHAR * pEventHost;
CHAR * pErrHdr =
    {"*** ERROR *** ERROR *** ERROR *** ERROR *** ERROR ***"};

INT WriteEventLog(CHAR * pDMsgs[],UINT uMsgCnt,UINT uSeverity);

//=====
//
// Function name: DiagIoInit
//
//=====
VOID DiagIoInit(CHAR * pDiagId,BOOL bConsole,BOOL bEvent,UINT uLevel)
{
    if (DIAGNOSTICS)
    {
        InitializeCriticalSection(&csDiagIo);

        uDiagLevel = uLevel;
        bEventLog = bEvent;
        bConsoleLog = bConsole;
    }
}

```

```

pEventHost = (CHAR *) malloc(10);
strcpy(pEventHost, ""); // local host
pDiagHdr = (CHAR *) malloc(strlen(pDiagId) + 1);
strcpy(pDiagHdr, pDiagId);
if (bEventLog)
{
    hEventLog = RegisterEventSource(pEventHost, pDiagId);
    if (hEventLog == NULL)
    {
        bEventLog = FALSE;
        if (bConsoleLog)
            fprintf(stdout,
                "%s: Event Log Register Failed (%ld)\n"
                "Event Log Will NOT be Used\n",
                pDiagHdr, GetLastError());
    }
    else
    {
        if (bConsoleLog)
            fprintf(stdout, "%s: Event Logging to LocalHost as %s\n",
                pDiagHdr, pDiagHdr);
    }
}; // if bEventLog
}; // if Diagnostics
}; // DiagIoInit

//=====
//
// Function name: DiagIoTerm
//
//=====
VOID DiagIoTerm(VOID)
{
    if (DIAGNOSTICS)
    {
        DeleteCriticalSection(&csDiagIo);
        if (hEventLog != NULL)
            DeregisterEventSource(hEventLog);
        free(pDiagHdr);
        free(pEventHost);
    }
}; // DiagIoTerm

//=====
//
// Function name: DiagIoWrite
//
//=====
VOID DiagIoWrite(CHAR * pDiagBuffer, UINT uSeverity)
{
    CHAR * pDMsgs[3];
    UINT uMsgCnt = 0;
    INT iERslt = 0;
    if (DIAGNOSTICS)
    {
        if (uDiagLevel >= uSeverity)
        {
            EnterCriticalSection(&csDiagIo);
            _try
            {

```

```

                if (uSeverity == DIAG_ERROR)
                {
                    pDMsgs[0] = pDiagHdr;
                    pDMsgs[1] = pErrHdr;
                    pDMsgs[2] = pDiagBuffer;
                    uMsgCnt = 3;
                }
                else
                {
                    pDMsgs[0] = pDiagHdr;
                    pDMsgs[1] = pDiagBuffer;
                    uMsgCnt = 2;
                }
            };
            if (bEventLog)
                iERslt = WriteEventLog(pDMsgs, uMsgCnt, uSeverity);
            if (bConsoleLog)
            {
                if (uMsgCnt == 3)
                    fprintf(stdout, "\n%s:
%s\n%s", pDMsgs[0], pDMsgs[1], pDMsgs[2]);
                else
                    fprintf(stdout, "\n%s: %s", pDMsgs[0], pDMsgs[1]);
                if (iERslt != 0)
                    fprintf(stdout,
                        "EventLog Write Failed (%ld), No Longer in Use\n",
                        iERslt);
            }
        };
    }
    _finally
    {
        LeaveCriticalSection(&csDiagIo);
    }
}; // if uDiagLevel >= uSeverity
}; // if Diagnostics
}; // DiagIoWrite

INT WriteEventLog(CHAR * pDMsgs[], UINT uMsgCnt, UINT uSeverity)
{
    WORD wType;
    WORD wCount;
    wCount = uMsgCnt;
    switch (uSeverity)
    {
        case DIAG_ERROR:
            wType = EVENTLOG_ERROR_TYPE;
            break;
        default:
            wType = EVENTLOG_INFORMATION_TYPE;
            break;
    }
};
if (wType != 0)
{
    if (!ReportEvent(hEventLog, // event log handle
                    wType, // event type
                    0, // category zero
                    uSeverity, // no event identifier
                    NULL, // no user security identifier
                    wCount, // # of substitution strings
                    0, // no binary data
                    (LPCTSTR *) pDMsgs, // address of string array
                    NULL)) // address of binary

```

```

    {
        DeregisterEventSource(hEventLog);
        hEventLog = NULL;
        bEventLog = FALSE;
        return(GetLastError());
    }; // ReportEvent failed
}; // if wType != 0
return(0);
}; // WriteEventLog

```

tpccproxy.def

```

LIBRARY      "tpcc_com_ps"

DESCRIPTION  'Proxy/Stub DLL'

EXPORTS
    DllGetClassObject      @1    PRIVATE
    DllCanUnloadNow        @2    PRIVATE
    DllRegisterServer      @3    PRIVATE
    DllUnregisterServer    @4    PRIVATE

```

tpccproxy.idl

```

// tpccproxy.idl
//
// Copyright Unisys, 1999
// Copyright Microsoft, 1999

// Forward declare all types defined
// Interface ITPCC;
import "oaidl.idl";
import "ocidl.idl";

[
    object,
    uuid(FEEE6AA2-84B1-11d2-BA47-00C04FBFE08B),
    helpstring("ITPCC Interface"),
    pointer_default(unique)
]
interface ITPCC : IUnknown
{
    HRESULT _stdcall NewOrder
    (
        [in, out] int * iSize,
        [in, out, size_is(*iSize)] char ** pData
    );

    HRESULT _stdcall Payment
    (
        [in, out] int * iSize,
        [in, out, size_is(*iSize)] char ** pData
    );

    HRESULT _stdcall Delivery

```

```

    (
        [in] int * iSize,
        [in, out, size_is(*iSize)] char ** pData
    );

    HRESULT _stdcall StockLevel
    (
        [in, out] int* iSize,
        [in, out, size_is(*iSize)] char ** pData
    );

    HRESULT _stdcall OrderStatus
    (
        [in, out] int* iSize,
        [in, out, size_is(*iSize)] char ** pData
    );

    HRESULT _stdcall CallSetComplete
    (
    );
}; // interface ITPCC

```

tpccsvr.def

; tpccsvr.def : Declares the module parameters.

```

LIBRARY      "tpcc_com_all.dll"

EXPORTS
    DllCanUnloadNow        @1    PRIVATE
    DllGetClassObject      @2    PRIVATE
    DllRegisterServer      @3    PRIVATE
    DllUnregisterServer    @4    PRIVATE

```

tpccsvr.idl

```

// tpccsvr.idl
//
// Copyright Unisys, 1999
// Copyright Microsoft, 1999

interface TPCC;

import "oaidl.idl";
import "ocidl.idl";
import "..\tpccproxy\tpccproxy.idl";

[
    uuid(122A3117-2520-11D3-BA71-00C04FBFE08B),
    version(1.0),
    helpstring("TPC-C 1.0 Type Library")
]
library TPCCLib
{
    importlib("stdole32.tlb");

```

```

importlib("stdole2.tlb");

[
    uuid(122A3128-2520-11D3-BA71-00C04FBFE08B),
    helpstring("All Txns Class")
]
coclass TPCC
{
    [default] interface ITPCC;
};

};

```

tpccsvr.h

```

// tpccsvr.h
//
// Copyright Unisys, 1999

class CTPCC_Common :
public ITPCC,
public IObjectControl,
public IObjectConstruct,
public CComObjectRootEx<CComSingleThreadModel>
{
public:
BEGIN_COM_MAP(CTPCC_Common)
    COM_INTERFACE_ENTRY(ITPCC)
    COM_INTERFACE_ENTRY(IObjectControl)
    COM_INTERFACE_ENTRY(IObjectConstruct)
END_COM_MAP()

CTPCC_Common();
~CTPCC_Common();

// ITPCC
public:
HRESULT __stdcall NewOrder(int * iSize, UCHAR ** pTData);
HRESULT __stdcall Payment(int * iSize, UCHAR ** pTData);
HRESULT __stdcall Delivery(int * iSize, UCHAR ** pTData);
HRESULT __stdcall StockLevel( int* iSize, UCHAR ** pTData);
HRESULT __stdcall OrderStatus(int* iSize, UCHAR ** pTData);
HRESULT __stdcall CallSetComplete();

// IObjectControl
STDMETHODIMP_(BOOL) CanBePooled() {return m_bCanBePooled;}
STDMETHODIMP Activate() {return S_OK;} // no transactions enlistment
STDMETHODIMP_(void) Deactivate() { }

// IObjectConstruct
STDMETHODIMP Construct(IDispatch * pUnk);

    // state
private:
    bool m_bCanBePooled;
    long m_lRefId;
    int m_iMaxRetry;
    int m_iTryCount;
    bool m_bTPRslt;

```

```

int m_iTPRslt;
union
{
    NEW_ORDER_DATA NOData;
    PAYMENT_DATA PYData;
    DELIVERY_DATA DLData;
    ORDER_STATUS_DATA OSData;
    STOCK_LEVEL_DATA SLData;
} m_TData;
SQLHDBC m_hdbc;
SQLHSTMT m_hstmt;
SQLHSTMT m_hstmtNewOrder;
SQLHSTMT m_hstmtPayment;
SQLHSTMT m_hstmtDelivery;
SQLHSTMT m_hstmtOrderStatus;
SQLHSTMT m_hstmtStockLevel;
SQLHDESC m_descNewOrderCols1;
SQLHDESC m_descNewOrderCols2;
SQLHDESC m_descOrderStatusCols1;
SQLHDESC m_descOrderStatusCols2;
SQLUIINTEGER m_BindOffset;
SQLUIINTEGER m_RowsFetched;
int m_no_commit_flag;

bool InitNOParams(void);
bool InitPYParams(void);
bool InitDLParams(void);
bool InitOSParams(void);
bool InitSLParams(void);
bool CheckDBError(char * pSource);
}; // Class CTPCC_Common

class CTPCC :
public CTPCC_Common,
public CComCoClass<CTPCC, &CLSID_TPCC>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_TPCC)

    BEGIN_COM_MAP(CTPCC)
        COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()
}; // Class CTPCC

```

tpccsvr.cpp

```

// tpccsvr.cpp
//
// Copyright Unisys, 1999
// Copyright Microsoft, 1999

#define STRICT
#define _WIN32_WINNT 0x0400
#define _ATL_APARTMENT_THREADED

#include <stdio.h>

```

```

#include <atlbase.h>
//You may derive a class from CComModule and use it if you want to
override
//something, but do not change the name of _Module
extern CComModule _Module;

#include <atlcom.h>
#include <initguid.h>
#include <transact.h>
#include <atlimpl.cpp>
#include <comsvcs.h>

#define DBNTWIN32
#include <sqltypes.h>
#include <sql.h>
#include <sqlext.h>
#include <odbcss.h>

#include "tpccproxy.h"
#include "tpcc.h"

#include "resource.h"
#include "tpccsvr_i.h"
#include "tpccsvr_i.c"
#include "tpccsvr.h"
#include "..\tpccproxy\tpccproxy_i.c"

CComModule _Module;

BEGIN_OBJECT_MAP(ObjectMap)
    OBJECT_ENTRY(CLSID_TPCC, CTPCC)
END_OBJECT_MAP()

char pProgId[100];
char * pPName = "TPCC Server";
int iInstance = 1;
char szServer[100] = "hostname";
char szMyHost[MAX_COMPUTERNAME_LENGTH + 1];
char szUser[100] = "sa";
char szPassword[100] = "";
char szDatabase[100] = "tpcc";
int iDeadlockRetry = 5;

static SQLHENV henv = SQL_NULL_HENV; // ODBC environment handle
long lCount = 0;
long lActive = 0;

bool ReadRegistry(VOID);
void WriteEventLog(char * pMsg,bool bError);

////////////////////////////////////
///
// DLL Entry Point

extern "C"
BOOL WINAPI DllMain(HINSTANCE hInst,ULONG ul_reason_for_call,LPVOID
lpReserved)
{
    char szDiag[300];
    DWORD dwCNSize = MAX_COMPUTERNAME_LENGTH;

```

```

try
{
    switch(ul_reason_for_call)
    {
        case DLL_PROCESS_ATTACH:
            _Module.Init(ObjectMap,hInst);
            DisableThreadLibraryCalls(hInst);
            wsprintf(pProgId,"%s%d",pPName,iInstance);
            if (ReadRegistry())
            {
                WriteEventLog("DllMain: Registry Key Not Present\n",TRUE);
                return(FALSE);
            };
            wsprintf(pProgId,"%s%d(%s)",pPName,iInstance,szServer);
            GetComputerName(szMyHost,&dwCNSize);
            szMyHost[dwCNSize] = 0;
            if (SQLAllocHandleStd(SQL_HANDLE_ENV,SQL_NULL_HANDLE,&henv) !=
SQL_SUCCESS)
            {
                wsprintf(szDiag,"DllMain(%s): Allocate Environment
Failed\n"
                "ServerName=%s,DB=%s,User=%s,PW=%s,Retries=%d\n",
                VERSIONINFO,szServer,szDatabase,szUser,szPassword,iDeadlockRetry);
                WriteEventLog(szDiag,FALSE);
                return FALSE;
            };
            wsprintf(szDiag,"DllMain(%s): Initialization Complete\n"
            "ServerName=%s,DB=%s,User=%s,PW=%s,Retries=%d\n",
            VERSIONINFO,szServer,szDatabase,szUser,szPassword,iDeadlockRetry);
            WriteEventLog(szDiag,FALSE);
            break;

            case DLL_PROCESS_DETACH:
                WriteEventLog("DllMain: Closing down for Process
Detach\n",FALSE);
                if (henv != NULL)
                    SQLFreeEnv(henv);
                _Module.Term();
                break;
            }; // switch ul_reason_for_call
    }
    catch (...)
    {
        wsprintf(szDiag,"DllMain: Unhandled exception during %s call\n",
        ul_reason_for_call == DLL_PROCESS_ATTACH ? "ATTACH" : "DETACH");
        WriteEventLog(szDiag,TRUE);
        return FALSE;
    };
    return TRUE;
}; // DllMain

STDAPI DllCanUnloadNow(void)
{
    return (_Module.GetLockCount()==0) ? S_OK : S_FALSE;
};

STDAPI DllGetClassObject(REFCLSID rclsid,REFIID riid,LPVOID* ppv)

```

```

{
    return _Module.GetClassObject(rclsid,riid,ppv);
};

STDAPI DllRegisterServer(void)
{
    // registers object, typelib and all interfaces in typelib
    return _Module.RegisterServer(TRUE);
};

STDAPI DllUnregisterServer(void)
{
    _Module.UnregisterServer();
    return S_OK;
};

inline void ReleaseInterface(IUnknown *pUnk)
{
    if (pUnk)
    {
        pUnk->Release();
        pUnk = NULL;
    };
};

CTPCC_Common::CTPCC_Common()
{
    char szDiag[300];
    Sleep(100);
    m_bCanBePooled = TRUE;
    m_iMaxRetry = iDeadlockRetry;
    m_lRefId = InterlockedIncrement(&lCount);
    InterlockedIncrement(&lActive);
    m_hdbc = SQL_NULL_HDBC;
    m_hstmt = SQL_NULL_HSTMT;
    m_hstmtNewOrder = SQL_NULL_HSTMT;
    m_hstmtPayment = SQL_NULL_HSTMT;
    m_hstmtDelivery = SQL_NULL_HSTMT;
    m_hstmtOrderStatus = SQL_NULL_HSTMT;
    m_hstmtStockLevel = SQL_NULL_HSTMT;
    m_descNewOrderCols1 = SQL_NULL_HDESC;
    m_descNewOrderCols2 = SQL_NULL_HDESC;
    m_descOrderStatusCols1 = SQL_NULL_HDESC;
    m_descOrderStatusCols2 = SQL_NULL_HDESC;
    wsprintf(szDiag,"CTPCC_Common: Initialized %ld\n",m_lRefId);
    WriteEventLog(szDiag,FALSE);
};

CTPCC_Common::~CTPCC_Common()
{
    char szDiag[300];
    SQLFreeHandle(SQL_HANDLE_STMT,m_hstmtNewOrder);
    SQLFreeHandle(SQL_HANDLE_STMT,m_hstmtPayment);
    SQLFreeHandle(SQL_HANDLE_STMT,m_hstmtDelivery);
    SQLFreeHandle(SQL_HANDLE_STMT,m_hstmtOrderStatus);
    SQLFreeHandle(SQL_HANDLE_STMT,m_hstmtStockLevel);
    SQLDisconnect(m_hdbc);
    SQLFreeHandle(SQL_HANDLE_DBC,m_hdbc);
    InterlockedDecrement(&lActive);
    wsprintf(szDiag,"~CTPCC_Common(%ld): Database closed\n",m_lRefId);
    WriteEventLog(szDiag,FALSE);
};

```

```

};

HRESULT CTPCC_Common::CallSetComplete()
{
    IObjectContext * pObjectContext = NULL;
    HRESULT hr = CoGetObjectContext( IID_IObjectContext, (void
**) &pObjectContext );
    pObjectContext->SetComplete();
    ReleaseInterface(pObjectContext);
    return hr;
};

bool CTPCC_Common::CheckDBError(char * pSource)
{
    RETCODE rc;
    SDWORD lNativeError;
    char szState[6];
    char szMsg[SQL_MAX_MESSAGE_LENGTH];
    char szTmp[7 * (SQL_MAX_MESSAGE_LENGTH + 2)];
    int iInx;
    int iMsgs = 0;
    bool bRslt = FALSE;
    iInx = wsprintf(szTmp,"%s",pSource);
    while (iMsgs < 6)
    {
        rc = SQLError(henv,m_hdbc,m_hstmt,(BYTE *) &szState,&lNativeError,
        (BYTE *) &szMsg,sizeof(szMsg),NULL);
        if (rc == SQL_NO_DATA)
            break;
        iMsgs++;
        // check for deadlock
        if (lNativeError == 1205)
        {
            iInx += wsprintf(szTmp + iInx,"Deadlock retry
(%d)\n",m_iTryCount);
            Sleep(10 * m_iTryCount);
            continue;
        };
        bRslt = TRUE;
        iInx += wsprintf(szTmp + iInx,"(%ld)%s\n",lNativeError,szMsg);
    }; // while iMsgs
    SQLFreeStmt(m_hstmt,SQL_CLOSE);
    WriteEventLog(szTmp,TRUE);
    return (bRslt);
}; // CheckDBError

STDMETHODIMP CTPCC_Common::Construct(IDispatch * pUnk)
{
    char szDiag[300];
    char szConnectStr[256];
    char szOutStr[1024];
    SQLSMALLINT iOutStrLen;
    char buffer[128];
    RETCODE rc;

    try
    {
        if (SQLAllocHandle(SQL_HANDLE_DBC,henv,&m_hdbc) != SQL_SUCCESS)
        {
            wsprintf(szDiag,"Construct(%ld): Allocate DBC Handle
failed\n",m_lRefId);

```



```

    CheckDBError(szDiag);
    return(E_FAIL);
};
if (SQLSetConnectOption(m_hdbc,SQL_PACKET_SIZE,4096) != SQL_SUCCESS)
{
    wsprintf(szDiag,
        "Construct(%ld): Set Connection Packet Size failed\n",
        m_lRefId);
    CheckDBError(szDiag);
    return(E_FAIL);
};
wsprintf(szConnectStr,
    "DRIVER=SQL Server;SERVER=%s;UID=%s;PWD=%s;DATABASE=%s;WSID=%s-
%ld",
    szServer,szUser,szPassword,szDatabase,szMyHost,m_lRefId);
rc = SQLDriverConnect(m_hdbc,NULL,(SQLCHAR * ) szConnectStr,
    sizeof(szConnectStr),(SQLCHAR * ) szOutStr, sizeof(szOutStr),
    &iOutStrLen,SQL_DRIVER_NOPROMPT);
if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
{
    wsprintf(szDiag,"Construct(%ld): Connect failed\n",m_lRefId);
    CheckDBError(szDiag);
    return(E_FAIL);
};
if (SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,&m_hstmt) != SQL_SUCCESS)
{
    wsprintf(szDiag,"Construct(%ld): Allocate STMT Handle
failed\n",m_lRefId);
    CheckDBError(szDiag);
    return(E_FAIL);
};

strcpy(buffer,"set nocount on set XACT_ABORT ON");
rc = SQLExecDirect(m_hstmt,(unsigned char *)buffer,SQL_NTS);
if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
{
    wsprintf(szDiag,"Construct(%ld): Set DB Options
failed\n",m_lRefId);
    CheckDBError(szDiag);
    return(E_FAIL);
};
SQLFreeHandle(SQL_HANDLE_STMT,m_hstmt);

if (InitNOParams())
    return(E_FAIL);
if (InitPYParams())
    return(E_FAIL);
if (InitDLParams())
    return(E_FAIL);
if (InitOSParams())
    return(E_FAIL);
if (InitSLParams())
    return(E_FAIL);
}
catch (...)
{
    wsprintf(szDiag,"Construct(%ld): Unhandled exception\n",m_lRefId);
    WriteEventLog(szDiag,TRUE);
    return(E_FAIL);
};

```

```

    wsprintf(szDiag,"Construct(%ld): Db connection
initialized\n",m_lRefId);
    WriteEventLog(szDiag,FALSE);
    return(S_OK);
}; // Construct

//=====================================================
// FUNCTION: UtilStrCpy
//
// Copies n characters from string pSrc to pDst and places a null
// null character at the end of the destination string. Unlike
// strncpy this function ensures that the result string is always
// null terminated.
//
//=====================================================
inline static void UtilStrCpy(char * pDest, const unsigned char * pSrc,
int n)
{
    strncpy(pDest,(char *)pSrc,n);
    pDest[n] = '\0';
    return;
}; // UtilStrCpy

bool CTPCC_Common::InitNOParams(void)
{
    char szDiag[300];
    int i;
    int j;
    if (SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,&m_hstmtNewOrder) !=
SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,m_hdbc,&m_descNewOrderCols1) !=
SQL_SUCCESS
        || SQLAllocHandle(SQL_HANDLE_DESC,m_hdbc,&m_descNewOrderCols2) !=
SQL_SUCCESS
        )
    {
        wsprintf(szDiag,"InitNOParams(%ld): Allocate Stmt Handles
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    m_hstmt = m_hstmtNewOrder;

    if (SQLSetStmtAttrW(m_hstmt,SQL_ATTR_APP_ROW_DESC,
        m_descNewOrderCols1,SQL_IS_POINTER) != SQL_SUCCESS)
    {
        wsprintf(szDiag,"InitNOParams(%ld): SetStmtAttr NOCols1
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    i = 0;
    if
(SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,
        0,0,&m_TData.NODATA.w_id,0,NULL) != SQL_SUCCESS
        ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,
        0,0,&m_TData.NODATA.d_id,0,NULL) != SQL_SUCCESS

```

```

    ||
SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_SLONG, SQL_INTEGER,
    0, 0, &m_TData.NOData.c_id, 0, NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_UTINYINT, SQL_TINYINT,
    0, 0, &m_TData.NOData.o_ol_cnt, 0, NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_UTINYINT, SQL_TINYINT,
    0, 0, &m_TData.NOData.o_all_local, 0, NULL) != SQL_SUCCESS
)
{
    wsprintf(szDiag, "InitNOParams(%ld): Bind Param failed\n", m_lRefId);
    CheckDBError(szDiag);
    return(TRUE);
};

for (j=0; j < MAX_OL; j++)
{
    if
(SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_SLONG, SQL_INTEGER,
    0, 0, &m_TData.NOData.OL[j].ol_i_id, 0, NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT,
    0, 0, &m_TData.NOData.OL[j].ol_supply_w_id, 0, NULL) !=
SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT,
    0, 0, &m_TData.NOData.OL[j].ol_quantity, 0, NULL) !=
SQL_SUCCESS
    )
    {
        wsprintf(szDiag, "InitNOParams(%ld): Bind Param(%d)
failed\n", m_lRefId, j);
        CheckDBError(szDiag);
        return(TRUE);
    };
};

// set the bind offset pointer
if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_ROW_BIND_OFFSET_PTR,
&m_BindOffset, SQL_IS_POINTER) != SQL_SUCCESS)
{
    wsprintf(szDiag, "InitNOParams(%ld): SetStmtAttr BindOffset
failed\n", m_lRefId);
    CheckDBError(szDiag);
    return(TRUE);
};

i = 0;
if (SQLBindCol(m_hstmt, ++i, SQL_C_CHAR, &m_TData.NOData.OL[0].ol_i_name,
    sizeof(m_TData.NOData.OL[0].ol_i_name), NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt, ++i, SQL_C_SSHORT, &m_TData.NOData.OL[0].ol_stock,
    0, NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt, ++i, SQL_C_CHAR, &m_TData.NOData.OL[0].ol_brand_generic,
    sizeof(m_TData.NOData.OL[0].ol_brand_generic), NULL) !=
SQL_SUCCESS
    ||
SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.NOData.OL[0].ol_i_price,
    0, NULL) != SQL_SUCCESS

```

```

    ||
SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.NOData.OL[0].ol_amount,
    0, NULL) != SQL_SUCCESS
)
{
    wsprintf(szDiag, "InitNOParams(%ld): Bind Col OL
failed\n", m_lRefId);
    CheckDBError(szDiag);
    return(TRUE);
};

// associate the column bindings for the second result set
if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
    m_descNewOrderCols2, SQL_IS_POINTER) != SQL_SUCCESS)
{
    wsprintf(szDiag, "InitNOParams(%ld): SetStmtAttr NOCols2
failed\n", m_lRefId);
    CheckDBError(szDiag);
    return(TRUE);
};

i = 0;
if (SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.NOData.w_tax,
    0, NULL) != SQL_SUCCESS
    || SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.NOData.d_tax,
    0, NULL) != SQL_SUCCESS
    || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG, &m_TData.NOData.o_id,
    0, NULL) != SQL_SUCCESS
    || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR, &m_TData.NOData.c_last,
    sizeof(m_TData.NOData.c_last), NULL) != SQL_SUCCESS
    || SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.NOData.c_discount,
    0, NULL) != SQL_SUCCESS
    || SQLBindCol(m_hstmt, ++i, SQL_C_CHAR, &m_TData.NOData.c_credit,
    sizeof(m_TData.NOData.c_credit), NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt, ++i, SQL_C_TYPE_TIMESTAMP, &m_TData.NOData.o_entry_d,
    0, NULL) != SQL_SUCCESS
    || SQLBindCol(m_hstmt, ++i, SQL_C_SLONG, &m_no_commit_flag,
    0, NULL) != SQL_SUCCESS
)
{
    wsprintf(szDiag, "InitNOParams(%ld): Bind Col failed\n", m_lRefId);
    CheckDBError(szDiag);
    return(TRUE);
};

return(FALSE);
}; // InitNOParams

HRESULT CTPCC_Common::NewOrder(int * iSize, UCHAR ** pData)
{
    NEW_ORDER_DATA * pnod;
    RETCODE rc;
    bool bRetry;

    // 0 1 2
    // 012345678901234567890123456789
    wchar_t szSqlTemplate[] = L"{call tpcc_neworder(?,?,?,?,?,
L"?,?,?,?,?,?,?,?,?,?,?,?,?",
L"?,?,?,?,?,?,?,?,?,?,?,?,?",
L"?,?,?,?,?,?,?,?,?,?,?,?,?)";

    int i;

```

```

pnod = (NEW_ORDER_DATA *) *pTData;
try
{
    m_hstmt = m_hstmtNewOrder;
    m_bTPRslt = TRUE;
    m_iTPRslt = SVCERR_DEADLOCK;
    // associate the parameter and column bindings for this transaction
    if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
        m_descNewOrderCols1, SQL_IS_POINTER ) != SQL_SUCCESS)
    {
        char szDiag[100];
        wsprintf(szDiag, "NewOrder(%ld): SetStmtAttr NOCols1\n", m_lRefId);
        CheckDBError(szDiag);
        pnod->bTPRslt = TRUE;
        pnod->iTPRslt = SVCERR_ODBC;
        strcpy(pnod->execution_status, "Error, Bind NOCols1 Failed");
        return(S_OK);
    };
    // clip statement buffer based on number of parameters
    // fixed part is 29 chars and variable part is 6 chars per line
item
    i = 29 + pnod->o_ol_cnt * 6;
    wcscpy(&szSqlTemplate[i], L"}");
    // check whether any order lines are for a remote warehouse
    pnod->o_all_local = 1;
    for (i = 0; i < pnod->o_ol_cnt; i++)
    {
        if (pnod->Ol[i].ol_supply_w_id != pnod->w_id)
        {
            pnod->o_all_local = 0;
            break;
        };
    };
    memcpy(&m_TData.NOData, pnod, sizeof(NEW_ORDER_DATA));
    for (m_iTryCount = 1; m_iTryCount <= m_iMaxRetry; m_iTryCount++)
    {
        bRetry = FALSE;
        m_BindOffset = 0;
        rc = SQLExecDirectW(m_hstmt, (SQLWCHAR *) szSqlTemplate, SQL_NTS);
        if (rc != SQL_SUCCESS && rc !=
SQL_SUCCESS_WITH_INFO)
        {
            char szDiag[100];
            wsprintf(szDiag, "NewOrder(%ld): Execute Direct\n", m_lRefId);
            if (CheckDBError(szDiag))
            {
                m_iTPRslt = SVCERR_ODBC;
                break;
            };
            continue;
        };
        // Get order line results
        m_TData.NOData.total_amount = 0;
        for (i = 0; i < m_TData.NOData.o_ol_cnt; i++)
        {
            // set the bind offset value...
            m_BindOffset = i * sizeof(m_TData.NOData.Ol[0]);
            if (SQLFetch(m_hstmt) == SQL_ERROR)
            {
                char szDiag[100];

```

```

        wsprintf(szDiag, "NewOrder(%ld): Fetch Ol%d\n", m_lRefId, (i +
1));
        if (CheckDBError(szDiag))
            m_iTPRslt = SVCERR_ODBC;
        else
            bRetry = TRUE;
        break;
    };
    // move to the next resultset
    if (SQLMoreResults(m_hstmt) == SQL_ERROR)
    {
        char szDiag[100];
        wsprintf(szDiag, "NewOrder(%ld): MoreResults
Ol%d\n", m_lRefId, (i + 1));
        if (CheckDBError(szDiag))
            m_iTPRslt = SVCERR_ODBC;
        else
            bRetry = TRUE;
        break;
    };
    m_TData.NOData.total_amount += m_TData.NOData.Ol[i].ol_amount;
}; // for o_ol_cnt
if (m_iTPRslt == SVCERR_ODBC)
    break;
if (bRetry)
    continue;
// associate the column bindings for the second result set
if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
    m_descNewOrderCols2, SQL_IS_POINTER ) != SQL_SUCCESS)
{
    char szDiag[100];
    wsprintf(szDiag, "NewOrder(%ld): SetStmtAttr
NOCols2\n", m_lRefId);
    if (CheckDBError(szDiag))
    {
        m_iTPRslt = SVCERR_ODBC;
        break;
    };
    continue;
};
if (SQLFetch(m_hstmt) == SQL_ERROR)
{
    char szDiag[100];
    wsprintf(szDiag, "NewOrder(%ld): Fetch\n", m_lRefId);
    if (CheckDBError(szDiag))
    {
        m_iTPRslt = SVCERR_ODBC;
        break;
    };
    continue;
};
SQLFreeStmt(m_hstmt, SQL_CLOSE);
if (m_no_commit_flag == 1)
{
    m_TData.NOData.total_amount *=
        ((1 + m_TData.NOData.w_tax + m_TData.NOData.d_tax) *
        (1 - m_TData.NOData.c_discount));
    m_bTPRslt = FALSE;
    m_iTPRslt = SVC_NOERROR;
}
}
else

```

```

        m_iTPRslt = SVC_BADITEMID;
        break;
    }; // for m_iMaxRetry
    memcpy(pnod,&m_TData.NOData,sizeof(NEW_ORDER_DATA));
    pnod->bTPRslt = m_bTPRslt;
    pnod->iTPRslt = m_iTPRslt;
    if (!m_bTPRslt)
    {
        strcpy(pnod->execution_status,"Transaction committed.");
        return(S_OK);
    };
    if (m_iTPRslt == SVC_BADITEMID)
    {
        strcpy(pnod->execution_status,"Item number is not valid.");
        return(S_OK);
    };
    if (m_iTPRslt == SVCERR_DEADLOCK)
        sprintf(pnod->execution_status,"Hit retry
max(%d).",m_iMaxRetry);
    else
        strcpy(pnod->execution_status,"DBTranAbort, Check Input Data");
    return(S_OK);
}

catch (...)
{
    char szDiag[300];
    sprintf(szDiag,"NewOrder(%ld): Unhandled exception\n",m_lRefId);
    WriteEventLog(szDiag,TRUE);
    m_bCanBePooled = FALSE;
    pnod->bTPRslt = TRUE;
    pnod->iTPRslt = SVCERR_EXCEPTION;
    strcpy(pnod->execution_status,"Error, Unhandled exception");
    return(S_OK);
};

}; // NewOrder

bool CTPCC_Common::InitPYParams(void)
{
    int i;
    char szDiag[300];
    if (SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,&m_hstmtPayment) !=
SQL_SUCCESS)
    {
        sprintf(szDiag,"InitPYParams(%ld): Allocate Stmt Handle
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };
    m_hstmt = m_hstmtPayment;
    i = 0;
    if
(SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,
0,0,&m_TData.PYData.w_id,0,NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,
0,0,&m_TData.PYData.c_w_id,0,NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_DOUBLE,SQL_NUMERIC,
6,2,&m_TData.PYData.h_amount,0,NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,

```

```

0,0,&m_TData.PYData.d_id, 0, NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,
0,0,&m_TData.PYData.c_d_id,0,NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,
0,0,&m_TData.PYData.c_id,0,NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_CHAR,SQL_CHAR,
sizeof(m_TData.PYData.c_last),0,&m_TData.PYData.c_last,
sizeof(m_TData.PYData.c_last), NULL) != SQL_SUCCESS
    )
    {
        sprintf(szDiag,"InitPYParams(%ld): Bind Param
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };
    i = 0;
    if (SQLBindCol(m_hstmt,++i,SQL_C_SLONG,&m_TData.PYData.
c_id,0,NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_last,
sizeof(m_TData.PYData.c_last), NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_TYPE_TIMESTAMP,&m_TData.PYData.h_date,
0,NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.w_street_1,
sizeof(m_TData.PYData.w_street_1), NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.w_street_2,
sizeof(m_TData.PYData.w_street_2),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.w_city,
sizeof(m_TData.PYData.w_city),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.w_state,
sizeof(m_TData.PYData.w_state),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.w_zip,
sizeof(m_TData.PYData.w_zip),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.d_street_1,
sizeof(m_TData.PYData.d_street_1),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.d_street_2,
sizeof(m_TData.PYData.d_street_2),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.d_city,
sizeof(m_TData.PYData.d_city),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.d_state,
sizeof(m_TData.PYData.d_state),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.d_zip,
sizeof(m_TData.PYData.d_zip),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_first,
sizeof(m_TData.PYData.c_first),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_middle,
sizeof(m_TData.PYData.c_middle),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_street_1,
sizeof(m_TData.PYData.c_street_1),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_street_2,
sizeof(m_TData.PYData.c_street_2),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_city,
sizeof(m_TData.PYData.c_city),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_state,
sizeof(m_TData.PYData.c_state),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_zip,
sizeof(m_TData.PYData.c_zip),NULL) != SQL_SUCCESS
    ||
SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.PYData.c_phone,
sizeof(m_TData.PYData.c_phone),NULL) != SQL_SUCCESS

```

```

||
SQLBindCol(m_hstmt, ++i, SQL_C_TYPE_TIMESTAMP, &m_TData.PYData.c_since,
0, NULL) != SQL_SUCCESS
|| SQLBindCol(m_hstmt, ++i, SQL_C_CHAR, &m_TData.PYData.c_credit,
sizeof(m_TData.PYData.c_credit), NULL) != SQL_SUCCESS
|| SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.PYData.c_credit_lim,
0, NULL) != SQL_SUCCESS
|| SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.PYData.c_discount,
0, NULL) != SQL_SUCCESS
|| SQLBindCol(m_hstmt, ++i, SQL_C_DOUBLE, &m_TData.PYData.c_balance,
0, NULL) != SQL_SUCCESS
|| SQLBindCol(m_hstmt, ++i, SQL_C_CHAR, &m_TData.PYData.c_data,
sizeof(m_TData.PYData.c_data), NULL) != SQL_SUCCESS
)
{
    wsprintf(szDiag, "InitPYParams(%ld): Bind Col failed\n", m_lRefId);
    CheckDBError(szDiag);
    return(TRUE);
};
return(FALSE);
}; // InitPYParams

HRESULT CTPCC_Common::Payment(int * iSize, UCHAR ** pTData)
{
    PAYMENT_DATA * ppd;
    RETCODE rc;
    ppd = (PAYMENT_DATA *) *pTData;
    try
    {
        m_hstmt = m_hstmtPayment;
        m_bTPRslt = TRUE;
        m_iTPRslt = SVCERR_DEADLOCK;
        if (ppd->c_id != 0)
            ppd->c_last[0] = 0;
        memcpy(&m_TData.PYData, ppd, sizeof(PAYMENT_DATA));
        for (m_iTryCount = 1; m_iTryCount <= m_iMaxRetry; m_iTryCount++)
        {
            rc = SQLExecDirectW(m_hstmt, (SQLWCHAR *)L"call
tpcc_payment(?,?,?,?,,?)", SQL_NTS);
            if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
            {
                char szDiag[100];
                wsprintf(szDiag, "Payment(%ld): Execute Direct\n", m_lRefId);
                if (CheckDBError(szDiag))
                {
                    m_iTPRslt = SVCERR_ODBC;
                    break;
                };
                continue;
            };
            if (SQLFetch(m_hstmt) == SQL_ERROR)
            {
                char szDiag[100];
                wsprintf(szDiag, "Payment(%ld): Fetch\n", m_lRefId);
                if (CheckDBError(szDiag))
                {
                    m_iTPRslt = SVCERR_ODBC;
                    break;
                };
                continue;
            };
        };
    };
};

```

```

SQLFreeStmt(m_hstmt, SQL_CLOSE);
if (m_TData.PYData.c_id == 0)
    m_iTPRslt = SVCERR_NOCUSTOMER;
else
{
    m_bTPRslt = FALSE;
    m_iTPRslt = SVC_NOERROR;
}
break;
}; // for m_iMaxRetry
memcpy(ppd, &m_TData.PYData, sizeof(PAYMENT_DATA));
ppd->bTPRslt = m_bTPRslt;
ppd->iTPRslt = m_iTPRslt;
if (!m_bTPRslt)
{
    strcpy(ppd->execution_status, "Transaction committed.");
    return(S_OK);
};
if (m_iTPRslt == SVCERR_NOCUSTOMER)
{
    strcpy(ppd->execution_status, "Invalid Customer id,name.");
    return(S_OK);
};
if (m_iTPRslt == SVCERR_DEADLOCK)
    wsprintf(ppd->execution_status, "Hit retry max(%d).", m_iMaxRetry);
else
    strcpy(ppd->execution_status, "DBTranAbort, Check Input Data");
return(S_OK);
}

catch (...)
{
    char szDiag[300];
    wsprintf(szDiag, "Payment(%ld): Unhandled exception\n", m_lRefId);
    WriteEventLog(szDiag, TRUE);
    m_bCanBePooled = FALSE;
    ppd->bTPRslt = TRUE;
    ppd->iTPRslt = SVCERR_EXCEPTION;
    strcpy(ppd->execution_status, "Error, Unhandled exception");
    return(S_OK);
};
}; // Payment

bool CTPCC_Common::InitDLParams(void)
{
    int i;
    char szDiag[300];
    if (SQLAllocHandle(SQL_HANDLE_STMT, m_hdbc, &m_hstmtDelivery) !=
SQL_SUCCESS)
    {
        wsprintf(szDiag, "InitDLParams(%ld): Allocate Stmt Handle
failed\n", m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };
    m_hstmt = m_hstmtDelivery;
    i = 0;
    if
(SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT,
0, 0, &m_TData.DLData.w_id, 0, NULL) != SQL_SUCCESS
||
SQLBindParameter(m_hstmt, ++i, SQL_PARAM_INPUT, SQL_C_SSHORT, SQL_SMALLINT,

```

```

        0,0,&m_TData.DLData.o_carrier_id,0,NULL) != SQL_SUCCESS
    )
    {
        wsprintf(szDiag,"InitDLParams(%ld): Bind Param
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };
    for (i = 0; i < 10; i++)
    {
        if
(SQLBindCol(m_hstmt, (UWORD)(i+1), SQL_C_SLONG, &m_TData.DLData.o_id[i],
0, NULL) != SQL_SUCCESS)
        {
            wsprintf(szDiag,"InitPYParams(%ld): Bind Col (%d)
failed\n",m_lRefId,(i+1));
            CheckDBError(szDiag);
            return(TRUE);
        };
    };
    return(FALSE);
}; // InitDLParams

HRESULT CTPCC_Common::Delivery(int * iSize, UCHAR ** pTData)
{
    DELIVERY_DATA * pdd;
    RETCODE rc;
    pdd = (DELIVERY_DATA *) *pTData;
    try
    {
        m_hstmt = m_hstmtDelivery;
        m_bTPRslt = TRUE;
        m_iTPRslt = SVCERR_DEADLOCK;
        memcpy(&m_TData.DLData,pdd,sizeof(DELIVERY_DATA));
        for (m_iTryCount = 1; m_iTryCount <= m_iMaxRetry; m_iTryCount++)
        {
            rc = SQLExecDirectW(m_hstmt, (SQLWCHAR *)L"call
tpcc_delivery(?,?)", SQL_NTS);
            if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
            {
                char szDiag[100];
                wsprintf(szDiag,"Delivery(%ld): Execute Direct\n",m_lRefId);
                if (CheckDBError(szDiag))
                {
                    m_iTPRslt = SVCERR_ODBC;
                    break;
                };
                continue;
            };
            if ( SQLFetch(m_hstmt) == SQL_ERROR )
            {
                char szDiag[100];
                wsprintf(szDiag,"Delivery(%ld): Fetch\n",m_lRefId);
                if (CheckDBError(szDiag))
                {
                    m_iTPRslt = SVCERR_ODBC;
                    break;
                };
                continue;
            };
            SQLFreeStmt(m_hstmt, SQL_CLOSE);

```

```

        m_bTPRslt = FALSE;
        m_iTPRslt = SVC_NOERROR;
        break;
    }; // for m_iMaxRetry
    memcpy(pdd,&m_TData.DLData,sizeof(DELIVERY_DATA));
    pdd->bTPRslt = m_bTPRslt;
    pdd->iTPRslt = m_iTPRslt;
    return(S_OK);
}

catch (...)
{
    char szDiag[300];
    wsprintf(szDiag,"Delivery(%ld): Unhandled exception\n",m_lRefId);
    WriteEventLog(szDiag,TRUE);
    m_bCanBePooled = FALSE;
    pdd->bTPRslt = TRUE;
    pdd->iTPRslt = SVCERR_EXCEPTION;
    return(S_OK);
}; // Delivery

bool CTPCC_Common::InitOSParams(void)
{
    char szDiag[300];
    int i;
    if (SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,&m_hstmtOrderStatus) !=
SQL_SUCCESS
    || SQLAllocHandle(SQL_HANDLE_DESC,m_hdbc,&m_descOrderStatusCols1) !=
SQL_SUCCESS
    || SQLAllocHandle(SQL_HANDLE_DESC,m_hdbc,&m_descOrderStatusCols2) !=
SQL_SUCCESS
    )
    {
        wsprintf(szDiag,"InitOSParams(%ld): Allocate Stmt Handles
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    m_hstmt = m_hstmtOrderStatus;

    if (SQLSetStmtAttrW(m_hstmt,SQL_ATTR_APP_ROW_DESC,
m_descOrderStatusCols1,SQL_IS_POINTER) != SQL_SUCCESS)
    {
        wsprintf(szDiag,"InitOSParams(%ld): SetStmtAttr OSCols1
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    i = 0;
    if
(SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,
0,0,&m_TData.OSData.w_id,0,NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,
0,0,&m_TData.OSData.d_id,0,NULL) != SQL_SUCCESS
    ||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SLONG,SQL_INTEGER,
0,0,&m_TData.OSData.c_id,0,NULL) != SQL_SUCCESS
    || SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_CHAR,SQL_CHAR,

```

```

        sizeof(m_TData.OSData.c_last),0,&m_TData.OSData.c_last,
        sizeof(m_TData.OSData.c_last),NULL) != SQL_SUCCESS
    )
    {
        wsprintf(szDiag,"InitOSParams(%ld): Bind Param failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    // configure block cursor
    if (SQLSetStmtAttrW(m_hstmt,SQL_ATTR_ROW_BIND_TYPE,
        (SQLPOINTER)sizeof(m_TData.OSData.OlOrderStatusData[0]),0) !=
SQL_SUCCESS
        || SQLSetStmtAttrW(m_hstmt,SQL_ATTR_ROWS_FETCHED_PTR,
        &m_RowsFetched,0) != SQL_SUCCESS
    )
    {
        wsprintf(szDiag,"InitOSParams(%ld): SetStmtAttr RowsFetched
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    i = 0;
    if (SQLBindCol(m_hstmt,++i,SQL_C_SSHORT,
        &m_TData.OSData.OlOrderStatusData[0].ol_supply_w_id, 0, NULL)
!= SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_SLONG,
        &m_TData.OSData.OlOrderStatusData[0].ol_i_id, 0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_SSHORT,
        &m_TData.OSData.OlOrderStatusData[0].ol_quantity, 0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_DOUBLE,
        &m_TData.OSData.OlOrderStatusData[0].ol_amount, 0, NULL) !=
SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_TYPE_TIMESTAMP,
        &m_TData.OSData.OlOrderStatusData[0].ol_delivery_d, 0, NULL)
!= SQL_SUCCESS
    )
    {
        wsprintf(szDiag,"InitOSParams(%ld): Bind Col Ol
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    if (SQLSetStmtAttrW(m_hstmt,SQL_ATTR_APP_ROW_DESC,
        m_descOrderStatusCols2,SQL_IS_POINTER) != SQL_SUCCESS)
    {
        wsprintf(szDiag,"InitOSParams(%ld): SetStmtAttr OSCols2
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };

    i = 0;
    if (SQLBindCol(m_hstmt,++i,SQL_C_SLONG,&m_TData.OSData.c_id,
        0,NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.OSData.c_last,
        sizeof(m_TData.OSData.c_last),NULL) != SQL_SUCCESS

```

```

        || SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.OSData.c_first,
        sizeof(m_TData.OSData.c_first),NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_CHAR,&m_TData.OSData.c_middle,
        sizeof(m_TData.OSData.c_middle), NULL) != SQL_SUCCESS
    )
    {
        SQLBindCol(m_hstmt,++i,SQL_C_TYPE_TIMESTAMP,&m_TData.OSData.o_entry_d,
        0,NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_SSHORT,&m_TData.OSData.o_carrier_id,
        0,NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_DOUBLE,&m_TData.OSData.c_balance,
        0,NULL) != SQL_SUCCESS
        || SQLBindCol(m_hstmt,++i,SQL_C_SLONG,&m_TData.OSData.o_id,
        0,NULL) != SQL_SUCCESS
    )
    {
        wsprintf(szDiag,"InitOSParams(%ld): Bind Col failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };
    return(FALSE);
}; // InitOSParams

HRESULT CTPCC_Common::OrderStatus(int * iSize,UCHAR ** pTData)
{
    ORDER_STATUS_DATA * posd;
    RETCODE rc;

    posd = (ORDER_STATUS_DATA *) *pTData;
    try
    {
        m_hstmt = m_hstmtOrderStatus;
        m_bTPRslt = TRUE;
        m_iTPRslt = SVCERR_DEADLOCK;
        if (SQLSetStmtAttrW(m_hstmt,SQL_ATTR_APP_ROW_DESC,
            m_descOrderStatusCols1,SQL_IS_POINTER) != SQL_SUCCESS)
        {
            char szDiag[100];
            wsprintf(szDiag,"OrderStatus(%ld): SetStmtAttr
OSCols1\n",m_lRefId);
            CheckDBError(szDiag);
            posd->bTPRslt = TRUE;
            posd->iTPRslt = SVCERR_ODBC;
            strcpy(posd->execution_status,"Error, SetStmt OSCols1");
            return(S_OK);
        };
        if (posd->c_id != 0)
            posd->c_last[0] = 0;
        memcpy(&m_TData.OSData,posd,sizeof(ORDER_STATUS_DATA));
        for (m_iTryCount = 1; m_iTryCount <= m_iMaxRetry; m_iTryCount++)
        {
            // configure block cursor
            if
(SQLSetStmtAttrW(m_hstmt,SQL_ATTR_ROW_ARRAY_SIZE,(SQLPOINTER)1,0) !=
SQL_SUCCESS)
            {
                char szDiag[100];
                wsprintf(szDiag,"OrderStatus(%ld): SetStmtAttr
RowArraySize\n",m_lRefId);
                if (CheckDBError(szDiag))
                {
                    m_iTPRslt = SVCERR_ODBC;

```

```

        break;
    };
    continue;
};
rc = SQLExecDirectW(m_hstmt, (SQLWCHAR*)L"{call
tpcc_orderstatus(?,?,?,?)", SQL_NTS);
if ( ((rc == SQL_SUCCESS_WITH_INFO) && (m_RowsFetched != 0)) ||
(rc == SQL_ERROR) )
{
    char szDiag[100];
    wsprintf(szDiag, "OrderStatus(%ld): Execute
Direct\n", m_lRefId);
    if (CheckDBError(szDiag))
    {
        m_iTPRsIt = SVCERR_ODBC;
        break;
    };
    continue;
};
// configure block cursor
if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_ROW_ARRAY_SIZE,
(SQLPOINTER)MAX_OL, 0) != SQL_SUCCESS)
{
    char szDiag[100];
    wsprintf(szDiag, "OrderStatus(%ld): SetStmtAttr RowArraySize
MAX_OL\n", m_lRefId);
    if (CheckDBError(szDiag))
    {
        m_iTPRsIt = SVCERR_ODBC;
        break;
    };
    continue;
};
rc = SQLFetchScroll(m_hstmt, SQL_FETCH_NEXT, 0);
if ( ((rc == SQL_SUCCESS_WITH_INFO) && (m_RowsFetched != 0)) ||
(rc == SQL_ERROR) )
{
    char szDiag[100];
    wsprintf(szDiag, "OrderStatus(%ld): FetchScroll\n", m_lRefId);
    if (CheckDBError(szDiag))
    {
        m_iTPRsIt = SVCERR_ODBC;
        break;
    };
    continue;
};
m_TData.OSData.o_ol_cnt = (short)m_RowsFetched;

if (m_TData.OSData.o_ol_cnt != 0)
{
    if
(SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC, m_descOrderStatusCols2,
SQL_IS_POINTER) != SQL_SUCCESS)
    {
        char szDiag[100];
        wsprintf(szDiag, "OrderStatus(%ld): SetStmtAttr
OSCols2\n", m_lRefId);
        if (CheckDBError(szDiag))
        {
            m_iTPRsIt = SVCERR_ODBC;
            break;
        };
    };
}

```

```

};
continue;
};
if (SQLMoreResults(m_hstmt) == SQL_ERROR)
{
    char szDiag[100];
    wsprintf(szDiag, "OrderStatus(%ld):
MoreResults\n", m_lRefId);
    if (CheckDBError(szDiag))
    {
        m_iTPRsIt = SVCERR_ODBC;
        break;
    };
    continue;
};
if (SQLFetch(m_hstmt) == SQL_ERROR)
{
    char szDiag[100];
    wsprintf(szDiag, "OrderStatus(%ld): Fetch\n", m_lRefId);
    CheckDBError(szDiag);
    if (SQLSetStmtAttrW(m_hstmt, SQL_ATTR_APP_ROW_DESC,
m_descOrderStatusCols1, SQL_IS_POINTER) != SQL_SUCCESS)
    {
        char szDiag[100];
        wsprintf(szDiag, "OrderStatus(%ld): SetStmtAttr (Fetch
OSCols1\n", m_lRefId);
        CheckDBError(szDiag);
        posd->bTPRsIt = TRUE;
        posd->iTPRsIt = SVCERR_ODBC;
        strcpy(posd->execution_status, "Error, SetStmt Cols1
Fetch");
        return(S_OK);
    };
    memcpy(&m_TData.OSData, posd, sizeof(ORDER_STATUS_DATA));
    Sleep(20);
    continue;
};
}; // order lines exist
SQLFreeStmt(m_hstmt, SQL_CLOSE);
if (m_TData.OSData.o_ol_cnt == 0)
    m_iTPRsIt = SVCERR_NOORDERS;
else
if (m_TData.OSData.c_id == 0 && m_TData.OSData.c_last[0] == 0)
    m_iTPRsIt = SVCERR_NOCUSTOMER;
else
{
    m_bTPRsIt = FALSE;
    m_iTPRsIt = SVC_NOERROR;
};
break;
}; // for m_iMaxRetry
memcpy(posd, &m_TData.OSData, sizeof(ORDER_STATUS_DATA));
posd->bTPRsIt = m_bTPRsIt;
posd->iTPRsIt = m_iTPRsIt;
if (!m_bTPRsIt)
{
    strcpy(posd->execution_status, "Transaction committed.");
    return(S_OK);
};
if (m_iTPRsIt == SVCERR_NOCUSTOMER)
{

```



```

        strcpy(posd->execution_status,"Invalid Customer id,name.");
        return(S_OK);
    };
    if (m_iTPRslt == SVCERR_NOORDERS)
    {
        strcpy(posd->execution_status,"Customer has no orders.");
        return(S_OK);
    };
    if (m_iTPRslt == SVCERR_DEADLOCK)
        sprintf(posd->execution_status,"Hit retry
max(%d).",m_iMaxRetry);
    else
        strcpy(posd->execution_status,"DBTranAbort, Check Input Data");
        return(S_OK);
    }
    catch (...)
    {
        char szDiag[300];
        sprintf(szDiag,"OrderStatus(%ld): Unhandled exception\n",m_lRefId);
        WriteEventLog(szDiag,TRUE);
        m_bCanBePooled = FALSE;
        posd->bTPRslt = TRUE;
        posd->iTPRslt = SVCERR_EXCEPTION;
        strcpy(posd->execution_status,"Error, Unhandled exception");
        return(S_OK);
    };
}; // Orderstatus

bool CTPCC_Common::InitSLParams(void)
{
    char szDiag[300];
    int i = 0;
    if (SQLAllocHandle(SQL_HANDLE_STMT,m_hdbc,&m_hstmtStockLevel) !=
SQL_SUCCESS )
    {
        sprintf(szDiag,"InitSLParams(%ld): Allocate Stmt Handle
failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };
    m_hstmt = m_hstmtStockLevel;
    if
(SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,
0,0,&m_TData.SLData.w_id, 0, NULL) != SQL_SUCCESS
||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_UTINYINT,SQL_TINYINT,
0,0,&m_TData.SLData.d_id, 0, NULL) != SQL_SUCCESS
||
SQLBindParameter(m_hstmt,++i,SQL_PARAM_INPUT,SQL_C_SSHORT,SQL_SMALLINT,
0,0,&m_TData.SLData.thresh_hold, 0, NULL) != SQL_SUCCESS
)
    {
        sprintf(szDiag,"InitSLParams(%ld): Bind Param failed\n",m_lRefId);
        CheckDBError(szDiag);
        return(TRUE);
    };
    if (SQLBindCol(m_hstmt,1,SQL_C_SLONG,
&m_TData.SLData.low_stock,0,NULL) != SQL_SUCCESS)
    {
        sprintf(szDiag,"InitSLParams(%ld): Bind Column failed\n",m_lRefId);
        CheckDBError(szDiag);

```

```

        return(TRUE);
    };
    return(FALSE);
}; // InitSLParams

HRESULT CTPCC_Common::StockLevel(int * iSize,UCHAR ** pTData)
{
    STOCK_LEVEL_DATA * psld;
    RETCODE rc;
    psld = (STOCK_LEVEL_DATA *) *pTData;
    try
    {
        m_hstmt = m_hstmtStockLevel;
        m_bTPRslt = TRUE;
        m_iTPRslt = SVCERR_DEADLOCK;
        memcpy(&m_TData.SLData,psld,sizeof(STOCK_LEVEL_DATA));
        for (m_iTryCount = 1; m_iTryCount <= m_iMaxRetry; m_iTryCount++)
        {
            rc = SQLExecDirectW(m_hstmt,
(SQLWCHAR*)L"{call tpcc_stocklevel(?,?,?)}",SQL_NTS);
            if (rc != SQL_SUCCESS && rc != SQL_SUCCESS_WITH_INFO)
            {
                char szDiag[100];
                sprintf(szDiag,"StockLevel(%ld): Execute Direct\n",m_lRefId);
                if (CheckDBError(szDiag))
                {
                    m_iTPRslt = SVCERR_ODBC;
                    break;
                };
                continue;
            };
            if (SQLFetch(m_hstmt) == SQL_ERROR)
            {
                char szDiag[100];
                sprintf(szDiag,"StockLevel(%ld): Fetch\n",m_lRefId);
                if (CheckDBError(szDiag))
                {
                    m_iTPRslt = SVCERR_ODBC;
                    break;
                };
                continue;
            };
            SQLFreeStmt(m_hstmt,SQL_CLOSE);
            m_bTPRslt = FALSE;
            m_iTPRslt = SVC_NOERROR;
            break;
        }; // for m_iMaxRetry
        memcpy(psld,&m_TData.SLData,sizeof(STOCK_LEVEL_DATA));
        psld->bTPRslt = m_bTPRslt;
        psld->iTPRslt = m_iTPRslt;
        if (!m_bTPRslt)
        {
            strcpy(psld->execution_status,"Transaction committed.");
            return(S_OK);
        };
        if (m_iTPRslt == SVCERR_DEADLOCK)
            sprintf(psld->execution_status,"Hit retry
max(%d).",m_iMaxRetry);
        else
            strcpy(psld->execution_status,"DBTranAbort, Check Input Data");
        return(S_OK);
    }

```

```

}
    catch (...)
    {
        char szDiag[300];
        wsprintf(szDiag,"StockLevel(%ld): Unhandled exception\n",m_lRefId);
        WriteEventLog(szDiag,TRUE);
        m_bCanBePooled = FALSE;
        psld->bTPRsIt = TRUE;
        psld->iTPRsIt = SVCERR_EXCEPTION;
        strcpy(psld->execution_status,"Error, Unhandled exception");
        return(S_OK);
    };
}; // Stocklevel

//=====
//
// Function name: ReadRegistry
//
// Sets global operational parameters from registry if they exist.
// Otherwise, compiled in defaults apply.
//
// Result:
// FALSE Registry entry found
// TRUE Registry entry does not exist
//
//=====
bool ReadRegistry(VOID)
{
    HKEY hkTPCC;
    DWORD dwMax;
    DWORD dwRT;
    CHAR * pszEnvData;
    char szValue[100];
    if (RegOpenKeyEx(HKEY_LOCAL_MACHINE,"SOFTWARE\\Unisys\\TPCC",0,
        KEY_READ, &hkTPCC) != ERROR_SUCCESS )
        return(TRUE);
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"SERVERNAME",0,&dwRT,(BYTE *)
&szValue,&dwMax)
    == ERROR_SUCCESS)
        strcpy(szServer,szValue);
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"DATABASE",0,&dwRT,(BYTE *) &szValue,&dwMax)
    == ERROR_SUCCESS)
        strcpy(szDatabase,szValue);
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"USER",0,&dwRT,(BYTE *) &szValue,&dwMax)
    == ERROR_SUCCESS)
        strcpy(szUser,szValue);
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"PASSWORD",0,&dwRT,(BYTE *) &szValue,&dwMax)
    == ERROR_SUCCESS)
        strcpy(szPassword,szValue);
    dwMax = sizeof(szValue);
    if (RegQueryValueEx(hkTPCC,"MAXRETRY",0,&dwRT,(BYTE *) &szValue,&dwMax)
    == ERROR_SUCCESS )
        iDeadlockRetry = abs(atoi(szValue));
    RegCloseKey(hkTPCC);
    if ((pszEnvData = getenv("WEBINSTANCE")) != NULL)
        strcpy(szServer,pszEnvData);
    if ((pszEnvData = getenv("WEBDATABASE")) != NULL)

```

```

        strcpy(szDatabase,pszEnvData);
    if ((pszEnvData = getenv("WEBDRIVERNO")) != NULL)
        iInstance = abs(atoi(pszEnvData));

    return(FALSE);
}; // ReadRegistry

//=====
//
// Function name: WriteEventLog
//
//=====
void WriteEventLog(char * pMsg,bool bError)
{
    WORD wType;
    char szHeader[100];
    char * pDMsgs[2];
    HANDLE hEventLog = NULL;
    if (bError)
        wType = EVENTLOG_ERROR_TYPE;
    else
        wType = EVENTLOG_INFORMATION_TYPE;
    hEventLog = RegisterEventSource(NULL,pProgId);
    wsprintf(szHeader,"%s (%ld)\n",pProgId,GetCurrentThreadId());
    pDMsgs[0] = szHeader;
    pDMsgs[1] = pMsg;
    if (hEventLog != NULL)
    {
        ReportEvent(hEventLog, // event log handle
            wType, // event type
            0, // category zero
            0, // no event identifier
            NULL, // no user security identifier
            2, // # of substitution strings
            0, // no binary data
            (LPCTSTR *) pDMsgs, // address of string array
            NULL); // address of binary data
        DeregisterEventSource(hEventLog);
    };
    return;
}; // WriteEventLog

```

Appendix B - Database Design

Build Scripts

BACKUP.SQL

```
-- File:      BACKUP.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Creates backup of tpcc database

DBCC TRACEON (3111)
GO

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select "Start date:", convert(varchar(30),@startdate,9)

dump database tpcc to tpccback8b, tpccback8c, tpccback8d, tpccback8e,
tpccback8f, tpccback8g, tpccback8h, tpccback8i with init, stats = 1,
blocksize=65536

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

go
```

BACKUPDEVB.SQL

```
-- File:      BACKUPDEVB.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Creates tpcc database Backup Devices

use master
go

-- create backup devices

exec sp_addumpdevice
'disk','tpccback8b','C:\Bigdb\dbBackups\backup8b\tpccback1.dmp'
exec sp_addumpdevice
'disk','tpccback8c','C:\Bigdb\dbBackups\backup8c\tpccback2.dmp'
exec sp_addumpdevice
'disk','tpccback8d','C:\Bigdb\dbBackups\backup8d\tpccback3.dmp'
exec sp_addumpdevice
'disk','tpccback8e','C:\Bigdb\dbBackups\backup8e\tpccback4.dmp'
```

```
exec sp_addumpdevice
'disk','tpccback8f','C:\Bigdb\dbBackups\backup8f\tpccback5.dmp'
exec sp_addumpdevice
'disk','tpccback8g','C:\Bigdb\dbBackups\backup8g\tpccback6.dmp'
exec sp_addumpdevice
'disk','tpccback8h','C:\Bigdb\dbBackups\backup8h\tpccback7.dmp'
exec sp_addumpdevice
'disk','tpccback8i','C:\Bigdb\dbBackups\backup8i\tpccback8.dmp'

go

CREATEDB.SQL
-- File:      CREATEDB.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.30
--           Copyright Microsoft, 1999
-- Purpose:   Creates tpcc database and backup files

use master
go

-- Create temporary table for timing

if exists ( select name from sysobjects where name = 'tpcc_timer' )
drop table tpcc_timer
go

create table tpcc_timer
(
    start_date          char(30),
    end_date            char(30)
)

insert into tpcc_timer values (0,0)
go

-- Store starting time

update tpcc_timer
set start_date = (select convert(char(30), getdate(),9))
go

-- create main database files

CREATE DATABASE tpcc
ON PRIMARY
(
    NAME          = MSSQL70_tpcc_root,
    FILENAME      = "C:\Bigdb\dbRoots\tpcc_root.mdf",
    SIZE          = 10MB,
    FILEGROWTH    = 0),
FILEGROUP MSSQL70_cs_fg
(
    NAME          = MSSQL70_csI8,
    FILENAME      = "C:\Bigdb\cabi\cs8",
    SIZE          = 7300MB,
    FILEGROWTH    = 0),
```

```

( NAME = MSSQL70_csH8,
  FILENAME = "C:\Bigdb\cabH\cs8\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG8,
  FILENAME = "C:\Bigdb\cabG\cs8\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF8,
  FILENAME = "C:\Bigdb\cabF\cs8\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csE8,
  FILENAME = "C:\Bigdb\cabE\cs8\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csD8,
  FILENAME = "C:\Bigdb\cabD\cs8\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csC8,
  FILENAME = "C:\Bigdb\cabC\cs8\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csB8,
  FILENAME = "C:\Bigdb\cabB\cs8\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csI7,
  FILENAME = "C:\Bigdb\cabI\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csH7,
  FILENAME = "C:\Bigdb\cabH\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG7,
  FILENAME = "C:\Bigdb\cabG\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF7,
  FILENAME = "C:\Bigdb\cabF\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csE7,
  FILENAME = "C:\Bigdb\cabE\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csD7,
  FILENAME = "C:\Bigdb\cabD\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csC7,
  FILENAME = "C:\Bigdb\cabC\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csB7,
  FILENAME = "C:\Bigdb\cabB\cs7\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csI6,

```

```

  FILENAME = "C:\Bigdb\cabI\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csH6,
  FILENAME = "C:\Bigdb\cabH\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG6,
  FILENAME = "C:\Bigdb\cabG\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF6,
  FILENAME = "C:\Bigdb\cabF\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csE6,
  FILENAME = "C:\Bigdb\cabE\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csD6,
  FILENAME = "C:\Bigdb\cabD\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csC6,
  FILENAME = "C:\Bigdb\cabC\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csB6,
  FILENAME = "C:\Bigdb\cabB\cs6\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csI5,
  FILENAME = "C:\Bigdb\cabI\cs5\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csH5,
  FILENAME = "C:\Bigdb\cabH\cs5\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG5,
  FILENAME = "C:\Bigdb\cabG\cs5\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF5,
  FILENAME = "C:\Bigdb\cabF\cs5\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csE5,
  FILENAME = "C:\Bigdb\cabE\cs5\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csD5,
  FILENAME = "C:\Bigdb\cabD\cs5\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csC5,
  FILENAME = "C:\Bigdb\cabC\cs5\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csB5,
  FILENAME = "C:\Bigdb\cabB\cs5\",

```

```

SIZE = 7300MB,
FILEGROWTH = 0),
( NAME = MSSQL70_csP4,
  FILENAME = "C:\Bigdb\cabP\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csO4,
  FILENAME = "C:\Bigdb\cabO\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csN4,
  FILENAME = "C:\Bigdb\cabN\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csL4,
  FILENAME = "C:\Bigdb\cabL\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csK4,
  FILENAME = "C:\Bigdb\cabK\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csJ4,
  FILENAME = "C:\Bigdb\cabJ\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csI4,
  FILENAME = "C:\Bigdb\cabI\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csh4,
  FILENAME = "C:\Bigdb\cabH\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG4,
  FILENAME = "C:\Bigdb\cabG\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF4,
  FILENAME = "C:\Bigdb\cabF\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_cse4,
  FILENAME = "C:\Bigdb\cabE\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csd4,
  FILENAME = "C:\Bigdb\cabD\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csc4,
  FILENAME = "C:\Bigdb\cabC\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csb4,
  FILENAME = "C:\Bigdb\cabB\cs4\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csa4,
  FILENAME = "C:\Bigdb\cabA\cs4\",
  SIZE = 7300MB,

```

```

FILEGROWTH = 0),
( NAME = MSSQL70_csP3,
  FILENAME = "C:\Bigdb\cabP\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csO3,
  FILENAME = "C:\Bigdb\cabO\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csN3,
  FILENAME = "C:\Bigdb\cabN\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csL3,
  FILENAME = "C:\Bigdb\cabL\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csK3,
  FILENAME = "C:\Bigdb\cabK\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csJ3,
  FILENAME = "C:\Bigdb\cabJ\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csI3,
  FILENAME = "C:\Bigdb\cabI\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csh3,
  FILENAME = "C:\Bigdb\cabH\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG3,
  FILENAME = "C:\Bigdb\cabG\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF3,
  FILENAME = "C:\Bigdb\cabF\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_cse3,
  FILENAME = "C:\Bigdb\cabE\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csd3,
  FILENAME = "C:\Bigdb\cabD\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csc3,
  FILENAME = "C:\Bigdb\cabC\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csb3,
  FILENAME = "C:\Bigdb\cabB\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csa3,
  FILENAME = "C:\Bigdb\cabA\cs3\",
  SIZE = 7300MB,
  FILEGROWTH = 0),

```

```

( NAME = MSSQL70_csP2,
  FILENAME = "C:\Bigdb\cabP\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csO2,
  FILENAME = "C:\Bigdb\cabO\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csN2,
  FILENAME = "C:\Bigdb\cabN\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csL2,
  FILENAME = "C:\Bigdb\cabL\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csK2,
  FILENAME = "C:\Bigdb\cabK\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csJ2,
  FILENAME = "C:\Bigdb\cabJ\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csI2,
  FILENAME = "C:\Bigdb\cabI\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csH2,
  FILENAME = "C:\Bigdb\cabH\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG2,
  FILENAME = "C:\Bigdb\cabG\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF2,
  FILENAME = "C:\Bigdb\cabF\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csE2,
  FILENAME = "C:\Bigdb\cabE\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csD2,
  FILENAME = "C:\Bigdb\cabD\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csC2,
  FILENAME = "C:\Bigdb\cabC\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csB2,
  FILENAME = "C:\Bigdb\cabB\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csA2,
  FILENAME = "C:\Bigdb\cabA\cs2\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csP1,

```

```

  FILENAME = "C:\Bigdb\cabP\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csO1,
  FILENAME = "C:\Bigdb\cabO\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csN1,
  FILENAME = "C:\Bigdb\cabN\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csL1,
  FILENAME = "C:\Bigdb\cabL\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csK1,
  FILENAME = "C:\Bigdb\cabK\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csJ1,
  FILENAME = "C:\Bigdb\cabJ\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csI1,
  FILENAME = "C:\Bigdb\cabI\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csH1,
  FILENAME = "C:\Bigdb\cabH\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csG1,
  FILENAME = "C:\Bigdb\cabG\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csF1,
  FILENAME = "C:\Bigdb\cabF\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csE1,
  FILENAME = "C:\Bigdb\cabE\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csD1,
  FILENAME = "C:\Bigdb\cabD\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csC1,
  FILENAME = "C:\Bigdb\cabC\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csB1,
  FILENAME = "C:\Bigdb\cabB\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_csA1,
  FILENAME = "C:\Bigdb\cabA\cs1\",
  SIZE = 7300MB,
  FILEGROWTH = 0),
FILEGROUP MSSQL70_misc_fg
( NAME = MSSQL70_miscA1,

```

```

FILENAME = "C:\Bigdb\cabA\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscB1,
FILENAME = "C:\Bigdb\cabB\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscC1,
FILENAME = "C:\Bigdb\cabC\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscD1,
FILENAME = "C:\Bigdb\cabD\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscE1,
FILENAME = "C:\Bigdb\cabE\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscF1,
FILENAME = "C:\Bigdb\cabF\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscG1,
FILENAME = "C:\Bigdb\cabG\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscH1,
FILENAME = "C:\Bigdb\cabH\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscI1,
FILENAME = "C:\Bigdb\cabI\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscJ1,
FILENAME = "C:\Bigdb\cabJ\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscK1,
FILENAME = "C:\Bigdb\cabK\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscL1,
FILENAME = "C:\Bigdb\cabL\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscN1,
FILENAME = "C:\Bigdb\cabN\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscO1,
FILENAME = "C:\Bigdb\cabO\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscP1,
FILENAME = "C:\Bigdb\cabP\misc1",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscA2,
FILENAME = "C:\Bigdb\cabA\misc2",

```

```

SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscB2,
FILENAME = "C:\Bigdb\cabB\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscC2,
FILENAME = "C:\Bigdb\cabC\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscD2,
FILENAME = "C:\Bigdb\cabD\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscE2,
FILENAME = "C:\Bigdb\cabE\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscF2,
FILENAME = "C:\Bigdb\cabF\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscG2,
FILENAME = "C:\Bigdb\cabG\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscH2,
FILENAME = "C:\Bigdb\cabH\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscI2,
FILENAME = "C:\Bigdb\cabI\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscJ2,
FILENAME = "C:\Bigdb\cabJ\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscK2,
FILENAME = "C:\Bigdb\cabK\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscL2,
FILENAME = "C:\Bigdb\cabL\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscN2,
FILENAME = "C:\Bigdb\cabN\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscO2,
FILENAME = "C:\Bigdb\cabO\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscP2,
FILENAME = "C:\Bigdb\cabP\misc2",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscA3,
FILENAME = "C:\Bigdb\cabA\misc3",
SIZE = 3500MB,

```

```

FILEGROWTH = 0),
( NAME = MSSQL70_miscB3,
  FILENAME = "C:\Bigdb\cabB\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscC3,
  FILENAME = "C:\Bigdb\cabC\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscD3,
  FILENAME = "C:\Bigdb\cabD\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscE3,
  FILENAME = "C:\Bigdb\cabE\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscF3,
  FILENAME = "C:\Bigdb\cabF\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscG3,
  FILENAME = "C:\Bigdb\cabG\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscH3,
  FILENAME = "C:\Bigdb\cabH\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscI3,
  FILENAME = "C:\Bigdb\cabI\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscJ3,
  FILENAME = "C:\Bigdb\cabJ\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscK3,
  FILENAME = "C:\Bigdb\cabK\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscL3,
  FILENAME = "C:\Bigdb\cabL\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscN3,
  FILENAME = "C:\Bigdb\cabN\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscO3,
  FILENAME = "C:\Bigdb\cabO\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscP3,
  FILENAME = "C:\Bigdb\cabP\misc3\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscA4,
  FILENAME = "C:\Bigdb\cabA\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),

```

```

( NAME = MSSQL70_miscB4,
  FILENAME = "C:\Bigdb\cabB\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscC4,
  FILENAME = "C:\Bigdb\cabC\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscD4,
  FILENAME = "C:\Bigdb\cabD\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscE4,
  FILENAME = "C:\Bigdb\cabE\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscF4,
  FILENAME = "C:\Bigdb\cabF\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscG4,
  FILENAME = "C:\Bigdb\cabG\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscH4,
  FILENAME = "C:\Bigdb\cabH\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscI4,
  FILENAME = "C:\Bigdb\cabI\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscJ4,
  FILENAME = "C:\Bigdb\cabJ\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscK4,
  FILENAME = "C:\Bigdb\cabK\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscL4,
  FILENAME = "C:\Bigdb\cabL\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscN4,
  FILENAME = "C:\Bigdb\cabN\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscO4,
  FILENAME = "C:\Bigdb\cabO\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscP4,
  FILENAME = "C:\Bigdb\cabP\misc4\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscB5,
  FILENAME = "C:\Bigdb\cabB\misc5\",
  SIZE = 3500MB,
  FILEGROWTH = 0),
( NAME = MSSQL70_miscC5,

```



```

FILENAME = "C:\Bigdb\cabC\misc5\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscD5,
FILENAME = "C:\Bigdb\cabD\misc5\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscE5,
FILENAME = "C:\Bigdb\cabE\misc5\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscF5,
FILENAME = "C:\Bigdb\cabF\misc5\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscG5,
FILENAME = "C:\Bigdb\cabG\misc5\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscH5,
FILENAME = "C:\Bigdb\cabH\misc5\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscI5,
FILENAME = "C:\Bigdb\cabI\misc5\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscB6,
FILENAME = "C:\Bigdb\cabB\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscC6,
FILENAME = "C:\Bigdb\cabC\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscD6,
FILENAME = "C:\Bigdb\cabD\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscE6,
FILENAME = "C:\Bigdb\cabE\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscF6,
FILENAME = "C:\Bigdb\cabF\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscG6,
FILENAME = "C:\Bigdb\cabG\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscH6,
FILENAME = "C:\Bigdb\cabH\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscI6,
FILENAME = "C:\Bigdb\cabI\misc6\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscB7,
FILENAME = "C:\Bigdb\cabB\misc7\",

```

```

SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscC7,
FILENAME = "C:\Bigdb\cabC\misc7\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscD7,
FILENAME = "C:\Bigdb\cabD\misc7\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscE7,
FILENAME = "C:\Bigdb\cabE\misc7\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscF7,
FILENAME = "C:\Bigdb\cabF\misc7\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscG7,
FILENAME = "C:\Bigdb\cabG\misc7\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscH7,
FILENAME = "C:\Bigdb\cabH\misc7\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscI7,
FILENAME = "C:\Bigdb\cabI\misc7\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscB8,
FILENAME = "C:\Bigdb\cabB\misc8\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscC8,
FILENAME = "C:\Bigdb\cabC\misc8\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscD8,
FILENAME = "C:\Bigdb\cabD\misc8\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscE8,
FILENAME = "C:\Bigdb\cabE\misc8\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscF8,
FILENAME = "C:\Bigdb\cabF\misc8\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscG8,
FILENAME = "C:\Bigdb\cabG\misc8\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscH8,
FILENAME = "C:\Bigdb\cabH\misc8\",
SIZE = 3500MB,
FILEGROWTH = 0),
(
NAME = MSSQL70_miscI8,
FILENAME = "C:\Bigdb\cabI\misc8\",
SIZE = 3500MB,

```

```

        FILEGROWTH      = 0)
LOG ON
(
    NAME                =MSSQL70_tpcc_log1,
    FILENAME             ="C:\Bigdb\dbLogs\tpcclog1\",
    SIZE                 =100001MB,
    FILEGROWTH          =0),
(
    NAME                =MSSQL70_tpcc_log2,
    FILENAME             ="C:\Bigdb\dbLogs\tpcclog2\",
    SIZE                 =100001MB,
    FILEGROWTH          =0)
go

-- Store ending time
update tpcc_timer
set   end_date         = (select convert(char(30), getdate(),9))
go

select "Elapsed time (in seconds): ", datediff(second,(select start_date
from tpcc_timer),(select end_date from tpcc_timer))

--      remove temporary table

if exists ( select name from sysobjects where name = 'tpcc_timer' )
drop table tpcc_timer
go

```

DBOPT1.SQL

```

-- File:      DBOPT1.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Resets database options after data load

```

```

sp_dboption tpcc,'select into/bulkcopy',FALSE
GO

```

```

sp_dboption tpcc,'trunc. log on chkpt.',FALSE
GO

USE tpcc
GO

CHECKPOINT
GO

sp_configure 'allow updates',1
GO

RECONFIGURE WITH OVERRIDE
GO

DECLARE @msg          varchar(50)

IF (SELECT (SUBSTRING((SELECT @@version),1,25))) = 'Microsoft SQL Server
2000'
    BEGIN
        --
        --           OPTIONS FOR SQL SERVER 8.0
        -- Set option values for user-defined indexes --
        --
        SET      @msg      = ' '
        PRINT   @msg
        SET      @msg      = 'Setting SQL Server 8.0 indexoptions'
        PRINT   @msg
        SET      @msg      = ' '
        PRINT   @msg

        EXEC sp_indexoption 'customer', 'DisallowPageLocks',
TRUE
        EXEC sp_indexoption 'district', 'DisallowPageLocks',
TRUE
        EXEC sp_indexoption 'warehouse', 'DisallowPageLocks',
TRUE
        EXEC sp_indexoption 'stock', 'DisallowPageLocks',
TRUE
        EXEC sp_indexoption 'order_line', 'DisallowRowLocks',
TRUE
        EXEC sp_indexoption 'orders', 'DisallowRowLocks',
TRUE
        EXEC sp_indexoption 'new_order', 'DisallowRowLocks',
TRUE
        EXEC sp_indexoption 'item', 'DisallowRowLocks',
TRUE
        EXEC sp_indexoption 'item', 'DisallowPageLocks',
TRUE
    END
ELSE
    BEGIN
        --
        --           OPTIONS FOR SQL SERVER 7.0
        -- Set option values for user-defined indexes --
        --
        SET      @msg      = ' '
        PRINT   @msg
        SET      @msg      = 'Setting SQL Server 7.0 indexoptions'
    END

```

```

PRINT @msg
SET @msg = ' '
PRINT @msg --

EXEC sp_indexoption 'customer', 'AllowPageLocks',
FALSE
EXEC sp_indexoption 'district', 'AllowPageLocks',
FALSE
EXEC sp_indexoption 'warehouse', 'AllowPageLocks',
FALSE
EXEC sp_indexoption 'stock', 'AllowPageLocks',
FALSE
EXEC sp_indexoption 'order_line', 'AllowRowLocks',
FALSE
EXEC sp_indexoption 'orders', 'AllowRowLocks',
FALSE
EXEC sp_indexoption 'new_order', 'AllowRowLocks',
FALSE
EXEC sp_indexoption 'item', 'AllowRowLocks',
FALSE
EXEC sp_indexoption 'item', 'AllowPageLocks',
FALSE

END
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 updates',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

```

DBOPT3.SQL

```

use tpcc
go

EXEC sp_tableoption 'new_order', 'pintable', true
GO

```

REMOVEDB.SQL

```

-- File: REMOVEDB.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.21
-- Copyright Microsoft, 1999, 2000
-- Purpose: Removes tpcc database and backup files

```

```

use master
go

-- remove any existing database and backup files

```

```

exec sp_dbremove tpcc, dropdev
go

```

```

exec sp_dropdevice 'tpccback8b'
exec sp_dropdevice 'tpccback8c'
exec sp_dropdevice 'tpccback8d'
exec sp_dropdevice 'tpccback8e'
exec sp_dropdevice 'tpccback8f'
exec sp_dropdevice 'tpccback8g'
exec sp_dropdevice 'tpccback8h'
exec sp_dropdevice 'tpccback8i'

```

```
go
```

RESTORE.SQL

```

-- File: RESTORE.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.21
-- Copyright Microsoft, 1999, 2000
-- Purpose: Loads database backup from backup files

```

```

DBCC TRACEON (3111)
GO

```

```

declare @startdate datetime
declare @enddate datetime
select @startdate = getdate()
select 'Start date:', convert(varchar(30),@startdate,9)

```

```

load database tpcc from tpccback8b, tpccback8c, tpccback8d, tpccback8e,
tpccback8f, tpccback8g, tpccback8h, tpccback8i with replace, stats =
1,blocksize=65536
go
dbcc traceoff (3111)

select @enddate = getdate()
select 'End date: ', convert(varchar(30),@enddate,9)
select 'Elapsed time (in seconds): ', datediff(second, @startdate,
@enddate)

go
dbcc traceoff (3111)

```

SETUP.CMD

```

::@ECHO OFF

@ECHO *****
@ECHO *
@ECHO * Microsoft TPC-C V3 Benchmark Kit Ver. 4.21 *
@ECHO *
@ECHO *****

@if '%1'==' ' goto usage
@if '%2'==' ' goto usage
@if '%3'==' ' goto usage
@if not '%4'==' ' if not '%4' == 'normal' if not '%4' == 'scale_down' goto
usage

:: Cleanup any old .err files
@if exist logs\*.err del logs\*.err >nul

@if '%3'=='full' goto start
@if '%3'=='bulddb' goto bulddb
@if '%3'=='objects' goto objects
@if '%3'=='bulkload' goto bulkload
@if '%3'=='objectsfull' goto objects
@if '%3'=='bulkloadfull' goto bulkload
@if '%3'=='backup' goto backup
@if '%3'=='backupfull' goto backup
@if '%3'=='verifyload' goto verifyload
goto usage

:start
:: Cleanup the logs directory...
@if exist logs\version.log del logs\version.log
>nul
@if exist logs\db.log del logs\db.log >nul
@if exist logs\objects.log del logs\objects.log
>nul
@if exist logs\objects.log del logs\objects.log
>nul
@if exist logs\bulkload.log del logs\bulkload.log >nul
@if exist logs\backup.log del logs\backup.log
>nul

@isql -Usa -P -S%1 -Q"select @@version"
> logs\version.log

```

```

@isql -Usa -P -S%1 -Q"select getdate()"
>> logs\version.log

:Verify_Installation
@isql -Usa -P -S%1 -b -iscripts\utility\verify_msg.sql
>nul
@isql -Usa -P -S%1 -b -iscripts\utility\verify_sort.sql
>nul
@isql -Usa -P -S%1 -b -Q"ms_verify_sort" >nul
@if errorlevel 1 goto BAD_SORT

:bulddb
@if exist logs\db.log del logs\db.log
>nul
@ECHO Removing any existing TPCC database and backup devices...
@isql -Usa -P -S%1 -e < scripts\%2.war\database\removedb.sql
> logs\db.log
@ECHO Creating Backup Device(s)...
@isql -Usa -P -S%1 -e < scripts\%2.war\database\backupdev.sql
>> logs\db.log
@if errorlevel 1 goto CREATE_ERROR
@ECHO Building database files and database...
@isql -Usa -P -S%1 -b -e < scripts\%2.war\database\createdb.sql
>> logs\db.log
@if errorlevel 1 goto CREATE_ERROR
@ECHO Database build complete.
@if '%3'=='full' goto objects
goto end

:objects
@if exist logs\objects.log del logs\objects.log
>nul
@ECHO Creating TPC-C database tables...
@isql -Usa -P -S%1 -b -e < scripts\%2.war\ddl\tables.sql >
logs\objects.log
@if errorlevel 1 goto TABLES_ERROR
@ECHO Creating database objects...
@isql -Usa -P -S%1 -b -e < scripts\dml\neword.sql >>
logs\objects.log
@if errorlevel 1 goto NEWORDER_ERROR
@isql -Usa -P -S%1 -b -e < scripts\dml\payment.sql >>
logs\objects.log
@if errorlevel 1 goto PAYMENT_ERROR
@isql -Usa -P -S%1 -b -e < scripts\dml\ordstat.sql >>
logs\objects.log
@if errorlevel 1 goto ORDERSTATUS_ERROR
@isql -Usa -P -S%1 -b -e < scripts\dml\delivery.sql >>
logs\objects.log
@if errorlevel 1 goto DELIVERY_ERROR
@isql -Usa -P -S%1 -b -e < scripts\dml\stocklev.sql >>
logs\objects.log
@if errorlevel 1 goto STOCKLEVEL_ERROR
@isql -Usa -P -S%1 -e < scripts\dml\version.sql
>> logs\objects.log
@ECHO Database object creation complete.
@if '%3'=='full' goto bulkload
@if '%3'=='objectsfull' goto bulkload
goto end

:bulkload
@if exist logs\bulkload.log del logs\bulkload.log >nul

```

```

@ECHO Setting database options before load...
@isql -Usa -P -S%1 -b -e < scripts\utility\dbopt1.sql
    >> logs\bulkload.log
@if errorlevel 1 goto DBOPT1_ERROR
@ECHO Beginning data load and index creation...
@isql -Usa -P -S%1 -b -e < scripts\%2.war\ddl\idxhiscl.sql    >
logs\idxhiscl.log
@if '%4'==' ' loader\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -
dscripts\%2.war\ddl -c0
@if errorlevel 1 goto END
@if '%4'=='normal' loader\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -
dscripts\%2.war\ddl -c0
@if errorlevel 1 goto END
@if '%4'=='scale_down' loader\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -
dscripts\%2.war\ddl -c1
@if errorlevel 1 goto END
goto bulkloaddone
:bulkloaddone
@ECHO Setting database options after load...
@isql -Usa -P -S%1 -b -e < scripts\utility\dbopt2.sql
    >> logs\bulkload.log
@if errorlevel 1 goto DBOPT2_ERROR
@ECHO Data load and index creation complete.

@ECHO.
@ECHO Calculating initial database space usage...
@cd..\acid\space
@call space.cmd %1
@cd..\..\setup

@if '%3'=='full' goto backup
@if '%3'=='objectsfull' goto backup
@if '%3'=='bulkloadfull' goto backup
goto end

:backup
@if exist logs\backup.log del logs\backup.log
    >nul
@ECHO Backing up database...
@isql -Usa -P -S%1 -b -e < scripts\%2.war\database\backup.sql
    > logs\backup.log
@if errorlevel 1 goto BACKUP_ERROR
@ECHO Database backup complete.
@if '%3'=='full' goto verifyload
@if '%3'=='objectsfull' goto verifyload
@if '%3'=='bulkloadfull' goto verifyload
@if '%3'=='backupfull' goto verifyload
goto complete

:verifyload
@if exist logs\verifyload.log del logs\verifyload.log
    >nul
@Echo Verifying TPC-C database load...
@isql -Usa -P -S%1 -b -e < scripts\utility\verifytpccload.sql    >
logs\verifyload.log
@if errorlevel 1 goto VERIFY_ERROR
@ECHO Check logs\verifyload.log to verify database load.

@ECHO.
@ECHO Checking database build
@cd ..\acid

```

```

@call doall.cmd %1
@cd ..\setup

:complete
@ECHO *****
@ECHO *
@ECHO * Full TPC-C V3 build complete. Check logs directory for setup
errors. *
@ECHO *
@ECHO *****
goto end

:usage
@ECHO *****
@ECHO *
@ECHO * The TPC-C setup command file requires the following parameters: *
@ECHO *
@ECHO * setup SERVER NUMWAR BLDOPT VERSION DBTYPE
@ECHO *
@ECHO * SERVER = machine name of server (use "" for local server)
@ECHO * NUMWAR = number of warehouses
@ECHO * BLDOPT = full, buildddb, objects, objectsfull, bulkload,
@ECHO * bulkloadfull, backup, or backupfull
@ECHO * DBTYPE = normal or scale_down
@ECHO *
@ECHO * Note #1: the BLDOPT and VERSION parameters are case sensitive.
@ECHO *
@ECHO * Note #2: the DBTYPE is optional. If no DBTYPE is specified,
SETUP *
@ECHO * will default to NORMAL.
@ECHO *
@ECHO * Example:
@ECHO *
@ECHO * The following command would be used to build a complete 200
@ECHO * warehouse database on SQL Server 7.0 running on server
\\myserver. *
@ECHO *
@ECHO * SETUP myserver 200 full
@ECHO *
@ECHO * NOTE 1: This command file does a backup of the database by
default *
@ECHO * after the database build process is complete. If you do not wish
*
@ECHO * to make a backup (strongly discouraged), you must edit this file
*
@ECHO * and comment that section out. Also, if you need to run the
dbcheck *
@ECHO * and the dbtables scripts on the fresh database load for an audit,
*
@ECHO * you must either run them manually or edit this file to include
them. *
@ECHO *
@ECHO * NOTE 2: The TPC-C setup program supports both Intel and Alpha
@ECHO * systems. It queries the %PROCESSOR_ARCHITECTURE% environment
@ECHO * variable and runs the appropriate executables.
@ECHO *
@ECHO *****
@goto end

```

```

:CREATE_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error in the database/backup device creation.
@echo.
@echo Check your CREATEDB.SQL, BACKUPDEV.SQL, LOGS\DB.LOG, and the
@echo SQL Server errorlog (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:TABLES_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error in the table creation.
@echo.
@echo Verify that the FileGroup names specified in CREATEDB.SQL
@echo match those specified in SCRIPTS\DDL\TABLES.SQL.
@echo.
@goto END

:NEWORDER_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error in the creation of the New Order stored
procedure.
@echo.
@echo Check your LOGS\OBJECTS.LOG, SCRIPTS\DML\NEWORD.SQL and the
@echo SQL Server errorlog (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:PAYMENT_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error in the creation of the Payment stored procedure.
@echo.
@echo Check your LOGS\OBJECTS.LOG, SCRIPTS\DML\PAYMENT.SQL and the
@echo SQL Server errorlog (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:ORDERSTATUS_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error in the creation of the Order Status stored
procedure.
@echo.
@echo Check your LOGS\OBJECTS.LOG, SCRIPTS\DML\ORDSTAT.SQL and the
@echo SQL Server errorlog (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:DELIVERY_ERROR
@echo.
@echo BUILD ABORTED!
@echo.

```

```

@echo There was an error in the creation of the Delivery stored procedure.
@echo.
@echo Check your LOGS\OBJECTS.LOG, SCRIPTS\DML\DELIVERY.SQL and the
@echo SQL Server errorlog (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:STOCKLEVEL_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error in the creation of the Stock Level stored
procedure.
@echo.
@echo Check your LOGS\OBJECTS.LOG, SCRIPTS\DML\STOCKLEV.SQL and the
@echo SQL Server errorlog (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:DBOPT1_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error setting the database options before load.
@echo.
@echo Check your LOGS\OBJECTS.LOG and the SQL Server errorlog
@echo (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:DBOPT2_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error setting the database options after load.
@echo.
@echo Check your LOGS\OBJECTS.LOG and the SQL Server errorlog
@echo (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:BACKUP_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error backing up the database after load.
@echo.
@echo Check your LOGS\BACKUP.LOG and the SQL Server errorlog
@echo (MSSQL7\LOG\ERRORLOG) for details.
@echo.
@goto END

:VERIFY_ERROR
@echo.
@echo BUILD ABORTED!
@echo.
@echo There was an error performing TPC-C database verification.
@echo.
@echo Check your LOGS\VERIFYLOAD.LOG and the SQL Server errorlog
@echo (MSSQL7\LOG\ERRORLOG) for details.
@echo.

```

```

@goto END

:BAD_SORT
@echo.
@echo BUILD ABORTED!
@echo.
@echo Incorrect SQL Server Sort Order. For performance and compatibility
@echo issues, you must run SQL Server with the Binary Sort Order.
@echo.
@echo For SQL Server 7.0, please re-install and specify the Binary Sort
Order.
@echo.
@echo For SQL Server 8.0, please re-install and specify Latin1_General
with the Binary Option.
@echo.
@goto END
:end

echo on

```

VERIFY_MSG.SQL

```

exec sp_dropmessage 50003
exec sp_addmessage 50003, 1,"Incorrect Sort Order - Please re-install SQL
Server with the Binary Sort Order"

```

VERIFY_SORT.SQL

```

-- File:      VERIFY_SORT.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Verifies the Sort Order

if exists (select name from sysobjects where name = "ms_verify_sort" )
    drop procedure ms_verify_sort
go

create proc ms_verify_sort

as

declare @sort_order    int

-- get the sort order
select @sort_order    = (select value from sysconfigures where config =
'1123')

if (select @sort_order) <> 50
    RAISERROR (50003,11,1)
Go

```

VERIFYTPCCLOAD

```

-- File:      VERIFYTPCCLOAD.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Performs series of TPCC database checks to verify
--           that database load completed correctly

print " "
select convert(char(30), getdate(),9)
print " "

use tpcc
go

-- *****
-- Check rows per table from SYSINDEXES
-- *****

print 'WAREHOUSE TABLE'

select rowcnt
from sysindexes
where id = object_id("warehouse")
go

print 'DISTRICT TABLE = (10 * No of warehouses)'

select rowcnt
from sysindexes
where id =object_id("district")
go

print 'ITEM TABLE = 100,000'

select rowcnt
from sysindexes
where id =object_id("item")
go

print 'CUSTOMER TABLE = (30,000 * No of warehouses)'

select rowcnt
from sysindexes
where id =object_id("customer")
go

print 'ORDERS TABLE = (30,000 * No of warehouses)'

select rowcnt
from sysindexes
where id =object_id("orders")
go

print 'HISTORY TABLE = (30,000 * No of warehouses)'

select rowcnt
from sysindexes

```

```

where id      =object_id("history")
go

print 'STOCK TABLE = (100,000 * No of warehouses)'
```

```

select rowcnt
from sysindexes
where id      =object_id("stock")
go

print 'ORDER_LINE TABLE = (300,000 * No of warehouses some change)'
```

```

select rowcnt
from sysindexes
where id      =object_id("order_line")
go

print 'NEW_ORDER TABLE = (9000 * No of warehouses)'
```

```

select rowcnt
from sysindexes
where id      =object_id("new_order")
go

-- *****
-- Check indices
-- *****

print '*****Index Check*****'
```

```

use tpcc
go

sp_helpindex customer
go

sp_helpindex stock
go

sp_helpindex district
go

sp_helpindex item
go

sp_helpindex new_order
go

sp_helpindex orders
go

sp_helpindex order_line
go

sp_helpindex warehouse
go
```

Tables

IDXCUSCL.SQL

```

-- File:      IDXCUSCL.SQL
--            Microsoft TPC-C Benchmark Kit Ver. 4.21
--            Copyright Microsoft, 1999, 2000
-- 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 MSSQL70_cs_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. 4.21
--            Copyright Microsoft, 1999, 2000
-- 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_ncl' )
    drop index customer.customer_ncl

create unique nonclustered index customer_ncl on customer(c_w_id, c_d_id,
c_last, c_first, c_id)
-- with SORT_IN_TEMPDB
on MSSQL70_cs_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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 MSSQL70_misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

go

```

IDXHISCL.SQL

```

-- File:      IDXHISCL.SQL
-- Purpose:   Creates clustered index on history 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 = 'history_cl' )
    drop index history.history_cl

create unique clustered index history_cl on history(h_c_w_id, h_date,
h_c_d_id, h_c_id, h_amount)
with SORT_IN_TEMPDB
on MSSQL70_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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 MSSQL70_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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 MSSQL70_ord_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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_c1' )
    drop index order_line.order_line_c1

create unique clustered index order_line_c1 on order_line(ol_w_id,
ol_d_id, ol_o_id, ol_number)
on MSSQL70_ord_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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_c1' )
    drop index orders.orders_c1

```

```

create unique clustered index orders_c1 on orders(o_w_id, o_d_id, o_id)
on MSSQL70_ord_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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)
with pad_index, fillfactor=90,
SORT_IN_TEMPDB
on MSSQL70_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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 MSSQL70_cs_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. 4.21
--           Copyright Microsoft, 1999, 2000
-- 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 MSSQL70_misc_fg

select @enddate = getdate()
select "End date: ", convert(varchar(30),@enddate,9)
select "Elapsed time (in seconds): ", datediff(second, @startdate,
@enddate)

go

```

TABLES.SQL

```

-- File:      TABLES.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Creates TPC-C tables

use tpcc
go

--
-- Remove all existing TPC-C tables
--

```

```

if exists ( select name from sysobjects where name = 'warehouse' )
    drop table warehouse
go
if exists ( select name from sysobjects where name = 'district' )
    drop table district
go
if exists ( select name from sysobjects where name = 'customer' )
    drop table customer
go
if exists ( select name from sysobjects where name = 'history' )
    drop table history
go
if exists ( select name from sysobjects where name = 'new_order' )
    drop table new_order
go
if exists ( select name from sysobjects where name = 'orders' )
    drop table orders
go
if exists ( select name from sysobjects where name = 'order_line' )
    drop table order_line
go
if exists ( select name from sysobjects where name = 'item' )
    drop table item
go
if exists ( select name from sysobjects where name = 'stock' )
    drop table stock
go

--
-- Create new tables
--

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 MSSQL70_misc_fg
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 MSSQL70_misc_fg
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            text
) on MSSQL70_cs_fg
  textimage_on MSSQL70_misc_fg
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 MSSQL70_misc_fg
go

```

```

create table new_order
(
    no_o_id                int,
    no_d_id                tinyint,
    no_w_id                smallint
) on MSSQL70_ord_fg
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 MSSQL70_ord_fg
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 MSSQL70_ord_fg
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 MSSQL70_misc_fg
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 MSSQL70_cs_fg
go

```

Stored Procedures

DELIVERY.SQL

```

-- File:      DELIVERY.SQL
--            Microsoft TPC-C Benchmark Kit Ver. 4.21.000
--            Copyright Microsoft, 1999, 2000
-- Purpose:   Creates delivery transaction stored procedure
--

```

```

--      Interface Level: 4.10.000

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

NEWORD.SQL

-- File:      NEWORD.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.21.000

```

```

--          Copyright Microsoft, 1999, 2000
-- Purpose:  Creates new order transaction stored procedure
--
--          Interface Level: 4.10.000

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 @li_no
                              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
when 7 then @s_w_id7
when 8 then @s_w_id8
when 9 then @s_w_id9
when 10 then @s_w_id10
when 11 then @s_w_id11
when 12 then @s_w_id12
when 13 then @s_w_id13
when 14 then @s_w_id14
when 15 then @s_w_id15
end,

@li_qty = case @li_no
when 1 then @ol_qty1
when 2 then @ol_qty2
when 3 then @ol_qty3
when 4 then @ol_qty4
when 5 then @ol_qty5
when 6 then @ol_qty6
when 7 then @ol_qty7
when 8 then @ol_qty8
when 9 then @ol_qty9
when 10 then @ol_qty10
when 11 then @ol_qty11
when 12 then @ol_qty12
when 13 then @ol_qty13
when 14 then @ol_qty14
when 15 then @ol_qty15
end

-- 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

-- update stock values

update stock
set    s_ytd          = s_ytd + @li_qty,
       @s_quantity    = s_quantity - @li_qty +
       case when (s_quantity -
@s_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

-- if there actually is a stock (and item) with these ids, go to work

if (@@rowcount > 0)
begin

-- insert order_line data (using data from item and stock)

insert into order_line values(@o_id,
                              @d_id,
                              @w_id,
                              @li_no,
                              @li_id,
                              @li_s_w_id,
                              "dec 31, 1899",
                              @li_qty,
                              @i_price * @li_qty,
                              @s_dist)

-- send line-item data to client

select @i_name,
       @s_quantity,
       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 (or stock) 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. 4.21.000
--            Copyright Microsoft, 1999, 2000
-- Purpose:   Creates order status transaction stored procedure
--
--            Interface Level: 4.10.000

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. 4.21.000
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Creates payment transaction stored procedure
--

```

```

-- Interface Level: 4.10.000
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),
        @c_data     char(42),
        @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 (repeatableread)
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 (repeatableread)
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,
@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)

-- update customer info

update customer
set c_data = @c_data + substring(c_data, 1, 458),
@screen_data = @c_data + substring(c_data, 1, 158)

where c_id = @c_id and
c_w_id = @c_w_id and
c_d_id = @c_d_id

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. 4.21.000
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Creates stock level transaction stored procedure
--
--           Interface Level: 4.10.000

```

```

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

VERSION.SQL

```

-- File:      VERSION.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21.000
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Returns version level of TPC-C stored procs
-- Note:      Always update the return value of this proc for
--           any interface changes or "must have" bug fixes.

```

```

-- The value returned by this SP defines the "interface level",
-- which must match between the stored procs and the client code.
-- The interface level may be down rev from the current kit. This
-- indicates that the interface hasn't changed since that version.

```

```

use tpcc
go

```

```

if exists ( select name from sysobjects where name = "tpcc_version" )
    drop procedure tpcc_version

```

go

```

create proc tpcc_version
as
declare @version        char(8)

```

begin

```

    select @version = "4.10.000"
    select @version as "Version"

```

end

go

Loader Source

GETARGS.C

```

// File:      GETARGS.C
//           Microsoft TPC-C Kit Ver. 4.21
//           Copyright Microsoft, 1996, 1997, 1998, 1999,
//           2000
// Purpose:   Source file for command line processing

```

```

// Includes
#include "tpcc.h"

```

```

//=====

```

```

//
// Function name: GetArgsLoader
//
//=====
void GetArgsLoader(int argc, char **argv, TPCCLDR_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     = DEFPLDPACKSIZE;
    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 throught */
            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 (strcmp(ptr+2,"item") == 0)
                        pargs->table_item = TRUE;
                    else if (strcmp(ptr+2,"warehouse")
== 0)
                        pargs->table_warehouse =
TRUE;
                    else if (strcmp(ptr+2,"customer") ==
0)
                        pargs->table_customer = TRUE;
                    else if (strcmp(ptr+2,"orders") ==
0)
                        pargs->table_orders = TRUE;
                    else
                    {
                        printf("\nUnrecognized command");
                        GetArgsLoaderUsage();
                        exit(1);
                    }

                    break;
                }

            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
%d\n", (long) BATCH);
        printf("-p TDS packet size
%d\n", (long) DEF_LDPACKSIZE);
        printf("-f Loader Results Output Filename
%s\n", LOADER_RES_FILE);
        printf("-s Starting Warehouse
%d\n", (long) DEF_STARTING_WAREHOUSE);
        printf("-i Build Option (data = 0, data and index = 1)
%d\n", (long) BUILD_INDEX);
        printf("-o Cluster Index Build Order (before = 1, after = 0)
%d\n", (long) INDEX_ORDER);
        printf("-c Build Scaled Database (normal = 0, tiny = 1)
%d\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);
}

```

RANDOM.C

```

//      File:          RANDOM.C
//                      Microsoft TPC-C Kit Ver. 4.21
//                      Copyright Microsoft, 1996, 1997, 1998, 1999,
2000
//      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;
}

STRINGS.C

//      File:          STRINGS.C
//      Microsoft TPC-C Kit Ver. 4.21
//      Copyright Microsoft, 1996, 1997, 1998, 1999,
2000
//      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;
    char cc = 'a';
    static char chArray[] =
"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
    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++)
    {
        cc = chArray[RandomNumber(0, chArrayMax)];
        str[i] = cc;
    }
    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 prcentage is valid
    if ((percent < 0) || (percent > 100))
    {
        printf("MakeOrigianlAlphaString: 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;
}

```

TIME.C

```

// File:            TIME.C
//
//                Microsoft TPC-C Kit Ver. 4.21
//                Copyright Microsoft, 1996, 1997, 1998, 1999,
2000
// 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;
}

```

TPCC.H

```

// File:            TPCC.H
//
//                Microsoft TPC-C Kit Ver. 4.21
//                Copyright Microsoft, 1996, 1997, 1998, 1999,
2000
// Purpose:        Header file for TPC-C database loader

```

```

// Build number of TPC Benchmark Kit
#define TPCKIT_VER    "4.21"

```

```

// 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 DEFLDPACKSIZE         32768
#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. 4.21
// Copyright Microsoft, 1996, 1997, 1998, 1999,
2000
// 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);

void CheckSQL();
void CheckDataBase();

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
char           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          v_hdbc; // for SQL Server
version verification
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         v_hstmt; // for SQL Server
version verification
HSTMT         i_hstmt1;
HSTMT         w_hstmt1;
HSTMT         o_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;

TPCCLDR_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*****");
    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);

    // verify database and tables exist before attempting to load

```

```

CheckDataBase();

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 != SUCCEED)
        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 != SUCCEED)
            HandleErrorDBC(i_hdbc1);
    }

    rc = bcp_bind(i_hdbc1, (BYTE *) &i_id, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT4, 1);
    if (rc != SUCCEED)
        HandleErrorDBC(i_hdbc1);

    rc = bcp_bind(i_hdbc1, (BYTE *) &i_im_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(i_hdbc1);

    rc = bcp_bind(i_hdbc1, (BYTE *) i_name, 0, I_NAME_LEN, NULL, 0, 0,
3);
    if (rc != SUCCEED)
        HandleErrorDBC(i_hdbc1);

    rc = bcp_bind(i_hdbc1, (BYTE *) &i_price, 0, SQL_VARLEN_DATA,
NULL, 0, SQLFLT8, 4);
    if (rc != SUCCEED)
        HandleErrorDBC(i_hdbc1);
}

```

```

5);
rc = bcp_bind(i_hdbc1, (BYTE *) i_data, 0, I_DATA_LEN, NULL, 0, 0,
);
if (rc != SUCCEED)
    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 != SUCCEED)
        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("idxwarcl");

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 != SUCCEED)
    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 != SUCCEED)
        HandleErrorDBC(w_hdbc1);
}

rc = bcp_bind(w_hdbc1, (BYTE *) &w_id, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT2, 1);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_name, 0, W_NAME_LEN, NULL, 0, 0,
2);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_street_1, 0, ADDRESS_LEN, NULL,
0, 0, 3);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_street_2, 0, ADDRESS_LEN, NULL,
0, 0, 4);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_city, 0, ADDRESS_LEN, NULL, 0,
0, 5);

```

```

if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_state, 0, STATE_LEN, NULL, 0, 0,
6);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) w_zip, 0, ZIP_LEN, NULL, 0, 0, 7);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &w_tax, 0, SQL_VARLEN_DATA, NULL,
0, SQLFLT8, 8);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &w_ytd, 0, SQL_VARLEN_DATA, NULL,
0, SQLFLT8, 9);
if (rc != SUCCEED)
    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 != SUCCEED)
        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("idxwarcl");

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        bcphint[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(bcphint, "tablock, order (d_w_id, d_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 10));
        rc = bcp_control(w_hdbc1, BCPHINTS, (void*) bcphint);
        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 != SUCCEED)
                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    bcphint[128];

// Seed with unique number
seed(3);

// if build index before load...
if ((aptr->build_index == 1) && (aptr->index_order == 1))
    BuildIndex("idxstkcl");

sprintf(name, "%s..%s", aptr->database, "stock");

rc = bcp_init(w_hdbc1, name, NULL, "logs\\stock.err", DB_IN);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcphint, "tablock, order (s_i_id, s_w_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 100000));
    rc = bcp_control(w_hdbc1, BCPHINTS, (void*) bcphint);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);
}

rc = bcp_bind(w_hdbc1, (BYTE *) &s_i_id, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT4, 1);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

bcp_bind(w_hdbc1, (BYTE *) &s_w_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 2);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) &s_quantity, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 3);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_01, 0, S_DIST_LEN, NULL, 0,
0, 4);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_02, 0, S_DIST_LEN, NULL, 0,
0, 5);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_03, 0, S_DIST_LEN, NULL, 0,
0, 6);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_04, 0, S_DIST_LEN, NULL, 0,
0, 7);
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);

```

```

0, 8); rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_05, 0, S_DIST_LEN, NULL, 0,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
0, 9); rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_06, 0, S_DIST_LEN, NULL, 0,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
0, 10); rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_07, 0, S_DIST_LEN, NULL, 0,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
0, 11); rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_08, 0, S_DIST_LEN, NULL, 0,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
0, 12); rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_09, 0, S_DIST_LEN, NULL, 0,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
0, 13); rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_10, 0, S_DIST_LEN, NULL, 0,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
0, SQLINT4, 14); rc = bcp_bind(w_hdbc1, (BYTE *) &s_ytd, 0, SQL_VARLEN_DATA, NULL,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
NULL, 0, SQLINT2, 15); rc = bcp_bind(w_hdbc1, (BYTE *) &s_order_cnt, 0, SQL_VARLEN_DATA,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
NULL, 0, SQLINT2, 16); rc = bcp_bind(w_hdbc1, (BYTE *) &s_remote_cnt, 0, SQL_VARLEN_DATA,
if (rc != SUCCEED)
    HandleErrorDBC(w_hdbc1);
17); rc = bcp_bind(w_hdbc1, (BYTE *) s_data, 0, S_DATA_LEN, NULL, 0, 0,
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("idxstkcl");

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_1;
// 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);
    }
}

```

```

// Start parallel loading threads here...

// Start customer table thread
printf("...Loading customer table for: d_id = %d,
w_id = %d\n", d_id, w_id);
hThread[0] = CreateThread(NULL,
0,
(LPTHREAD_START_ROUTINE) LoadCustomerTable,
&customer_time_start,
0,
&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);
hThread[1] = CreateThread(NULL,
0,
(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(300, 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 != SUCCEED)
        HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_d_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_w_id, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT2, 3);
    if (rc != SUCCEED)
        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 != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 2);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 3);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_d_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 4);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_w_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 5);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

```

```

    rc = bcp_bind(c_hdbc2, (BYTE *) &h_date, 0, H_DATE_LEN, NULL, 0,
SQLCHARACTER, 6);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) &h_amount, 0, SQL_VARLEN_DATA, NULL,
0, SQLFLT8, 7);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = bcp_bind(c_hdbc2, (BYTE *) h_data, 0, H_DATA_LEN, NULL, 0, 0, 8);
    if (rc != SUCCEED)
        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 != SUCCEED)
            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                 w_id;
    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 != SUCCEED)
    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 != SUCCEED)
        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 != SUCCEED)
    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 != SUCCEED)
        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 != SUCCEED)
    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 != SUCCEED)
        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,
(LPCTSTR) 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,
(LPCTSTR) 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_DISTRICT+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_district);

    for (o_id=0;o_id<orders_per_district;o_id++)
    {
        // Generate ORDER and NEW-ORDER data
        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);

            // Generate ORDER-LINE data
            if (o_id < first_new_order)
            {
                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_delivery_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_delivery_d,"1899-12-31
00:00:00.000");
            }
        }
    }

//=====
//
// 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 != SUCCEED)
            HandleErrorDBC(o_hdbc1);

        rc = bcp_bind(o_hdbc1, (BYTE *) &o_entry_d, 0, O_ENTRY_D_LEN,
NULL, 0, SQLCHARACTER, 5);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc1);

        rc = bcp_bind(o_hdbc1, (BYTE *) &o_carrier_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 6);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc1);

        rc = bcp_bind(o_hdbc1, (BYTE *) &o_ol_cnt, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT2, 7);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc1);

        rc = bcp_bind(o_hdbc1, (BYTE *) &o_all_local, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 8);
        if (rc != SUCCEED)
            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 != SUCCEED)
            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("idxordcl");

        // 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 != SUCCEED)
        HandleErrorDBC(o_hdbc2);

```

```

        rc = bcp_bind(o_hdbc2, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 2);
        if (rc != SUCCEED)
            HandleErrorDBC(o_hdbc2);

        rc = bcp_bind(o_hdbc2, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 3);
        if (rc != SUCCEED)
            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 != SUCCEED)
                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("idxnodcl");
        }
    }
}

//=====
//
// 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].ol_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
(%d rows)\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 != SUCCEED)
        HandleErrorDBC(i_hdbc1);

    rc = SQLDriverConnect ( i_hdbc1,
                            NULL,

```

```

        (SQLCHAR*)&szDriverString[0]
        SQL_NTS,
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT );
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(w_hdbc1);
    rc = SQLDriverConnect ( w_hdbc1,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
);
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(c_hdbc1);
    rc = SQLDriverConnect ( c_hdbc1,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],

```

```

        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
);
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(c_hdbc2);
    rc = SQLDriverConnect ( c_hdbc2,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
);
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(o_hdbc1);
    rc = SQLDriverConnect ( o_hdbc1,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,

```

```

);
    SQL_DRIVER_NOPROMPT
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(o_hdbc2);

    rc = SQLDriverConnect ( o_hdbc2,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
);
    if (rc != SUCCEED)
        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 != SUCCEED)
        HandleErrorDBC(o_hdbc3);

    rc = SQLDriverConnect ( o_hdbc3,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
);
    if (rc != SUCCEED)

```

```

        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 HandleErrorSTMT (HSTMT hstmt1)
{
    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_STMT , hstmt1, 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;
}

//=====
//
// Function    : CheckDataBase
//
//=====

void CheckDataBase()
{
    RETCODE      rc;

    char         szDriverString[300];
    char         szDriverStringOut[1024];
    char         TablesBitMap[9] = {"000000000"};
    int          i, ExitFlag;

    SQLSMALLINT  cbDriverStringOut;
    SQLCHAR      TabName[10];
    SQLINTEGER    TabNameInd, TabCount, TabCountInd;

    ExitFlag = 0;

    SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE, &henv );

    SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION, (void*)SQL_OV_ODBC3, 0
);

    SQLAllocHandle(SQL_HANDLE_DBC, henv , &v_hdbc);

    SQLSetConnectAttr(v_hdbc, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );

    // Open connection to SQL Server

    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
            aptr->server,
            aptr->user,
            aptr->password,
            aptr->database );

    rc = SQLSetConnectAttr( v_hdbc, SQL_ATTR_PACKET_SIZE,
(SQLPOINTER)aptr->pack_size, SQL_IS_UIINTEGER );
    if (rc != SQL_SUCCESS)
        HandleErrorDBC(v_hdbc);

    rc = SQLDriverConnect ( v_hdbc,
                            NULL,
                            (SQLCHAR*)&szDriverString[0]
                            ,
                            SQL_NTS,

```

```

(SQLCHAR*)&szDriverStringOut[0],
                                sizeof(szDriverStringOut),
                                &cbDriverStringOut,
                                SQL_DRIVER_NOPROMPT );

// if the rc is SQL_ERROR, the the TPCC database probably does not
exist
if (rc == SQL_ERROR)
{
    printf("The database TPCC does not appear to exist!\n");
    printf("\nCheck LOGS\\ directory for database creation
errors.\n");

    // cleanup database connections and handles
    SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);
    SQLDisconnect(v_hdbc);
    SQLFreeHandle(SQL_HANDLE_DBC, v_hdbc);

    // since there is not a database, exit back to SETUP.CMD
    exit(1);
}

if ( SQLAllocHandle(SQL_HANDLE_STMT, v_hdbc , &v_hstmt) !=
SQL_SUCCESS )
    HandleErrorDBC(v_hdbc);

if ( SQLBindCol(v_hstmt, 1, SQL_C_ULONG, &TabCount, 0,
&TabCountInd) != SQL_SUCCESS )
    HandleErrorSTMT(v_hstmt);

// count the number of user tables from sysobjects
rc = SQLExecDirect(v_hstmt, "select count(*) from sysobjects where
xtype = \'U\'", SQL_NTS);
if ((rc != SQL_SUCCESS) && (rc != SQL_SUCCESS_WITH_INFO))
    HandleErrorSTMT(v_hstmt);

if ( SQLFetch(v_hstmt) != SQL_SUCCESS )
    HandleErrorSTMT(v_hstmt);

// if the number of tables is less than 9, select all the user
tables in TPCC
if (TabCount != 9)
{
    SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);

    SQLAllocHandle(SQL_HANDLE_STMT, v_hdbc , &v_hstmt);

    if ( SQLBindCol(v_hstmt, 1, SQL_C_CHAR, &TabName,
sizeof(TabName), &TabNameInd) != SQL_SUCCESS )
        HandleErrorSTMT(v_hstmt);

    // select the list of user tables into a result set
    rc = SQLExecDirect(v_hstmt, "select * from sysobjects where
xtype = \'U\'", SQL_NTS);
    if ((rc != SQL_SUCCESS) && (rc != SQL_SUCCESS_WITH_INFO))
        HandleErrorSTMT(v_hstmt);

    // go through the result set and set the bitmap for each
found table
    // set the bitmap to '1' if the table name is found

```

```

while ((rc = SQLFetch(v_hstmt)) != SQL_NO_DATA)
{
    switch( TabName[0] )
    {
        case 'w':
            TablesBitMap[0] = '1';
            break;
        case 'd':
            TablesBitMap[1] = '1';
            break;
        case 'c':
            TablesBitMap[2] = '1';
            break;
        case 'h':
            TablesBitMap[3] = '1';
            break;
        case 'n':
            TablesBitMap[4] = '1';
            break;
        case 'o':
            if (TabName[5] = 's')
                TablesBitMap[5] = '1';
            if (TabName[5] = '_')
                TablesBitMap[6] = '1';
            break;
        case 'i':
            TablesBitMap[7] = '1';
            break;
        case 's':
            TablesBitMap[8] = '1';
            break;
    }

    // a '0' ExitFlag means do NOT exit the loader early, a '1'
means exit the loader early
    ExitFlag = 0;

    // interate through the bitmap to display which table(s) is
actually missing
    for (i = 0; i <= 8; i++)
    {
        switch(i)
        {
            case 0:
                if (TablesBitMap[i] == '0')
                {
                    printf("The Warehouse table is
missing or damaged.\n");
                    ExitFlag = 1;
                }
                break;
            case 1:
                if (TablesBitMap[i] == '0')
                {
                    printf("The District table is
missing or damaged.\n");
                    ExitFlag = 1;
                }
                break;
        }
    }
}

```

```

        case 2:
            if (TablesBitMap[i] == '0')
            {
                printf("The Customer table is
missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 3:
            if (TablesBitMap[i] == '0')
            {
                printf("The History table is missing
or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 4:
            if (TablesBitMap[i] == '0')
            {
                printf("The New_Order table is
missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 5:
            if (TablesBitMap[i] == '0')
            {
                printf("The Orders table is missing
or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 6:
            if (TablesBitMap[i] == '0')
            {
                printf("The Order_Line table is
missing or damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 7:
            if (TablesBitMap[i] == '0')
            {
                printf("The Item table is missing or
damaged.\n");
                ExitFlag = 1;
            }
            break;
        case 8:
            if (TablesBitMap[i] == '0')
            {
                printf("The Stock table is missing
or damaged.\n");
                ExitFlag = 1;
            }
            break;
    }
}

```

```

// if one or more tables are missing, display message and
exit the loader

```

```

if (ExitFlag = 1)
{
    printf("\nExiting TPC-C Loader!\n");
    printf("\nCheck LOGS\\ directory for database\n");
    printf("or table creation errors.\n");

    // cleanup database connections and handles
    SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);
    SQLDisconnect(v_hdbc);
    SQLFreeHandle(SQL_HANDLE_DBC, v_hdbc);

    exit(1);
}

// cleanup database connections and handles
SQLFreeHandle(SQL_HANDLE_STMT, v_hstmt);
SQLDisconnect(v_hdbc);
SQLFreeHandle(SQL_HANDLE_DBC, v_hdbc);

return;

```


Appendix C - Tunable Parameters

Microsoft SQL Server Startup Parameters

```
c:\sql80\mssql\binn\sqlservr -c -x -g300 /T3502
```

Where:

- -c Start SQL Server independently of the Service Control Manager
- -x Disables the keeping of CPU time and cache hit ratio statistics
- /T3502 Writes a message to the SQL Server Errorlog showing the beginning and ending time of each checkpoint
- -g300 Specifies the amount of virtual address space, in MB, SQL Server will leave available for memory allocations, excluding the buffer pool and thread stacks, such as dynamically-loaded DLLs, extended procedure calls, etc. If this option is not specified, SQL Server will use a value that is suitable for a wide range of runtime environments. Use of this option may be appropriate in 2GB (3GB Enterprise Edition) configurations in which the memory usage requirements of SQL Server are atypical and the virtual address space of the SQL Server process is totally in use. Incorrect use of this option can lead to conditions under which SQL Server may not start or may encounter runtime errors.

BOOT.INI

The /pae switch was added to the boot.ini file to cause Windows 2000 to support more than 4GB of physical memory.

The following switch was added to bind interrupts from an IO module (DIB) to a specific 4-processor group:

```
/clstr=0706050403020100
```

User Rights Assignment

The Group Policy Editor of Windows 2000 was used to modify an entry under User Rights Assignment. Specifically, the right to "Lock pages in memory" was given to the

Administrators group so that SQL Server 2000 could use large amounts of physical memory.

Microsoft SQL Server Configuration Parameters

```
-- File:      VERSION.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Returns SQL Server version string
```

```
print " "
select convert(char(30), getdate(),9)
print " "
go
```

```
select @@version
go
```

```
1> 2> 3> 4> 5> 6> 7> 8> 9> 10> 11>
```

```
-----
Jul 19 2001  2:14:09:497PM
```

```
(1 row affected)
```

```
1> 2> 3>
```

```
-----
-----
-----
-----
-----
-----
Microsoft SQL Server 2000 - 8.00.391 (Intel X86)
Jul  9 2001 13:25:27
Cop
yright (c) 1988-2000 Microsoft Corporation
Enterprise Edition on Windo
ws NT 5.1 (Build 2509: )
```

```

(1 row affected)
1> 2>
-- File:      CONFIG.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.21
--           Copyright Microsoft, 1999, 2000
-- Purpose:   Collects SQL Server configuration parameters

print " "
select convert(char(30), getdate(),9)
print " "
go

sp_configure "show advanced",1
go
reconfigure with override
go
sp_configure
go
1> 2> 3> 4> 5> 6> 7> 8> 9> 10>

-----
Jul 19 2001  2:15:23:357PM

(1 row affected)

1> 2> 3> DBCC execution completed. If DBCC printed error messages, contact
your
system administrator.
Configuration option 'show advanced options' changed from 1 to 1. Run
the RECONFIGURE statement to install.
1> 2> 1> 2> name          minimum      maximum
config_value
run_value
-----
affinity mask            -2147483648  2147483647    -1
allow updates            0            1            1
awe enabled              0            1            1
c2 audit mode           0            1            0
cost threshold for parallelism 0            32767        1
cursor threshold        -1           2147483647   -1
default full-text language 0           2147483647  1033
default language         0            9999         0
fill factor (%)          0            100          0
index create memory (KB) 704          2147483647   0
lightweight pooling      0            1            1
locks                    5000         2147483647   0

```

```

0
max degree of parallelism 0            32            1
1
max server memory (MB)    4           2147483647   63500
63500
max text repl size (B)   0           2147483647   65536
65536
max worker threads       32          32767        512
512
media retention          0            365          0
0
min memory per query (KB) 512         2147483647   1024
1024
min server memory (MB)   0           2147483647   0
0
nested triggers         0            1            1
1
network packet size (B)  512         65536        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        119
119
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  600
600
scan for startup procs   0            1            0
0
set working set size     0            1            0
0
show advanced options    0            1            1
1
two digit year cutoff    1753        9999        2049
2049
user connections         0           32767        0
0
user options             0           32767        0
0
1>

```

ESM7700 Storage Configuration Information

Database DPE-Cache (common to 23 DPEs)

```
C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI3:0:0:0  
getcache
```

```
SP Read Cache State      Disabled  
SP Write Cache State     Disabled  
Cache Page size:        16  
Write Cache Mirrored:    YES  
Low Watermark:          15  
High Watermark:         85  
SPA Cache pages:        0  
SPB Cache pages:        0  
Unassigned Cache Pages: 0  
Read Hit Ratio:         0  
Write Hit Ratio:        0  
Prct Dirty Cache Pages = 0  
Prct Cache Pages Owned = 0  
SPA Read Cache State    Disabled  
SPB Read Cache State    Disabled  
SPA Write Cache State   Disabled  
SPB Write Cache State   Disabled  
SPS Test Day:          Sunday  
SPS Test Time:         01:00  
Physical memory size of CPU Ram = 512  
Free Memory Size of CPU Ram = 498  
SPA Read Cache Size = 0  
SPB Read Cache Size = 0  
SPA Write Cache Size = 0  
SPB Write Cache Size = 0  
SPA Optimized Raid 3 Memory Size = 0  
SPB Optimized Raid 3 Memory Size = 0
```

Database DPE LUN Configurations (common to 23 DPEs)

```
C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI3:0:0:0 getlun  
0
```

```
Prefetch size (blocks) = 4  
Prefetch multiplier = 4  
Segment size (blocks) = 4  
Segment multiplier = 4  
Maximum prefetch (blocks) = 512  
Prefetch Disable Size (blocks) = 129  
Prefetch idle count = 40
```

```
Prefetching: NO  
Prefetched data retained YES
```

```
Read cache configured according to  
specified parameters.
```

```
Minimum latency reads N/A
```

```
RAID Type:                RAID0  
RAIDGroup ID:             1  
State:                    Bound  
Stripe Crossing:         0  
Element Size:            128  
Current owner:           SP A  
Auto-trespass:           DISABLED  
Auto-assign:             DISABLED  
Write cache:             DISABLED  
Read cache:              DISABLED  
Idle Threshold:          0  
Idle Delay Time:         20  
Write Aside Size:        1023  
Default Owner:           SP A  
Rebuild Priority:         N/A  
Verify Priority:         N/A  
Prct Reads Forced Flushed: 0  
Prct Writes Forced Flushed: 0  
Prct Rebuilt:            100  
Prct Bound:              100  
Lun Capacity:            169362
```

```
Enclosure 0 Disk 0 Enabled
```

```
Enclosure 0 Disk 1 Enabled
```

```
Enclosure 0 Disk 2 Enabled
```

```
Enclosure 0 Disk 3 Enabled
```

```
Enclosure 0 Disk 4 Enabled
```

```
Enclosure 0 Disk 5 Enabled
```

```
Enclosure 0 Disk 6 Enabled
```

```
Enclosure 0 Disk 7 Enabled
```

```
Enclosure 0 Disk 8 Enabled
```

```
Enclosure 0 Disk 9 Enabled
```

```
Statistics logging is disabled.  
Certain fields are not printed if statistics  
logging is not enabled.
```

```
C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI3:0:1:0 getlun  
1
```

```
Prefetch size (blocks) = 4  
Prefetch multiplier = 4  
Segment size (blocks) = 4  
Segment multiplier = 4  
Maximum prefetch (blocks) = 512  
Prefetch Disable Size (blocks) = 129  
Prefetch idle count = 40
```

```
Prefetching: NO  
Prefetched data retained YES
```

Read cache configured according to specified parameters.

Minimum latency reads N/A

RAID Type: RAID0
RAIDGroup ID: 1
State: Bound
Stripe Crossing: 0
Element Size: 128
Current owner: SP A
Auto-trespass: DISABLED
Auto-assign: DISABLED
Write cache: DISABLED
Read cache: DISABLED
Idle Threshold: 0
Idle Delay Time: 20
Write Aside Size: 1023
Default Owner: SP A
Rebuild Priority: N/A
Verify Priority: N/A
Prct Reads Forced Flushed: 0
Prct Writes Forced Flushed: 0
Prct Rebuilt: 100
Prct Bound: 100
Lun Capacity: 169362

Enclosure 1 Disk 0 Enabled
Enclosure 1 Disk 1 Enabled
Enclosure 1 Disk 2 Enabled
Enclosure 1 Disk 3 Enabled
Enclosure 1 Disk 4 Enabled
Enclosure 1 Disk 5 Enabled
Enclosure 1 Disk 6 Enabled
Enclosure 1 Disk 7 Enabled
Enclosure 1 Disk 8 Enabled
Enclosure 1 Disk 9 Enabled

Statistics logging is disabled.
Certain fields are not printed if statistics logging is not enabled.

C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI3:0:2:0 getlun
2
Prefetch size (blocks) = 4
Prefetch multiplier = 4
Segment size (blocks) = 4
Segment multiplier = 4

Maximum prefetch (blocks) = 512
Prefetch Disable Size (blocks) = 129
Prefetch idle count = 40

Prefetching: NO
Prefetched data retained YES

Read cache configured according to specified parameters.

Minimum latency reads N/A

RAID Type: RAID0
RAIDGroup ID: 1
State: Bound
Stripe Crossing: 0
Element Size: 256
Current owner: SP B
Auto-trespass: DISABLED
Auto-assign: DISABLED
Write cache: DISABLED
Read cache: DISABLED
Idle Threshold: 0
Idle Delay Time: 20
Write Aside Size: 1023
Default Owner: SP A
Rebuild Priority: N/A
Verify Priority: N/A
Prct Reads Forced Flushed: 0
Prct Writes Forced Flushed: 0
Prct Rebuilt: 100
Prct Bound: 100
Lun Capacity: 169361

Enclosure 2 Disk 0 Enabled
Enclosure 2 Disk 1 Enabled
Enclosure 2 Disk 2 Enabled
Enclosure 2 Disk 3 Enabled
Enclosure 2 Disk 4 Enabled
Enclosure 2 Disk 5 Enabled
Enclosure 2 Disk 6 Enabled
Enclosure 2 Disk 7 Enabled
Enclosure 2 Disk 8 Enabled
Enclosure 2 Disk 9 Enabled

Statistics logging is disabled.
Certain fields are not printed if statistics logging is not enabled.


```
C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI3:0:3:0 getlun
3
```

```
Prefetch size (blocks) = 4
Prefetch multiplier = 4
Segment size (blocks) = 4
Segment multiplier = 4
Maximum prefetch (blocks) = 512
Prefetch Disable Size (blocks) = 129
Prefetch idle count = 40
```

```
Prefetching: NO
Prefetched data retained YES
```

```
Read cache configured according to
specified parameters.
```

```
Minimum latency reads N/A
```

```
RAID Type: RAID0
RAIDGroup ID: 1
State: Bound
Stripe Crossing: 0
Element Size: 256
Current owner: SP B
Auto-trespass: DISABLED
Auto-assign: DISABLED
Write cache: DISABLED
Read cache: DISABLED
Idle Threshold: 0
Idle Delay Time: 20
Write Aside Size: 1023
Default Owner: SP A
Rebuild Priority: N/A
Verify Priority: N/A
Prct Reads Forced Flushed: 0
Prct Writes Forced Flushed: 0
Prct Rebuilt: 100
Prct Bound: 100
Lun Capacity: 169361
```

```
Enclosure 3 Disk 0 Enabled
```

```
Enclosure 3 Disk 1 Enabled
```

```
Enclosure 3 Disk 2 Enabled
```

```
Enclosure 3 Disk 3 Enabled
```

```
Enclosure 3 Disk 4 Enabled
```

```
Enclosure 3 Disk 5 Enabled
```

```
Enclosure 3 Disk 6 Enabled
```

```
Enclosure 3 Disk 7 Enabled
```

```
Enclosure 3 Disk 8 Enabled
```

```
Enclosure 3 Disk 9 Enabled
```

```
Statistics logging is disabled.
Certain fields are not printed if statistics
logging is not enabled.
```

Log DPE-Cache Configuration

```
C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI2:0:1:0
getcache
```

```
SP Read Cache State Disabled
SP Write Cache State Enabled
Cache Page size: 16
Write Cache Mirrored: YES
Low Watermark: 15
High Watermark: 85
SPA Cache pages: 15605
SPB Cache pages: 15605
Unassigned Cache Pages: 0
Read Hit Ratio: 0
Write Hit Ratio: 0
Prct Dirty Cache Pages = 0
Prct Cache Pages Owned = 0
SPA Read Cache State Disabled
SPB Read Cache State Disabled
SPA Write Cache State Enabled
SPB Write Cache State Enabled
SPS Test Day: Sunday
SPS Test Time: 01:00
Physical memory size of CPU Ram = 512
Free Memory Size of CPU Ram = 0
SPA Read Cache Size = 0
SPB Read Cache Size = 0
SPA Write Cache Size = 498
SPB Write Cache Size = 498
SPA Optimized Raid 3 Memory Size = 0
SPB Optimized Raid 3 Memory Size = 0
```

Log DPE LUN Configurations

```
C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI2:0:1:0 getlun
0
```

```
Prefetch size (blocks) = 4
Prefetch multiplier = 4
Segment size (blocks) = 4
Segment multiplier = 4
Maximum prefetch (blocks) = 512
Prefetch Disable Size (blocks) = 129
Prefetch idle count = 40
```

```
Variable length prefetching YES
Prefetched data retained YES
```

```
Read cache configured according to
```

specified parameters.

Minimum latency reads N/A

```

RAID Type:                RAID1/0
RAIDGroup ID:             1
State:                    Bound
Stripe Crossing:          0
Element Size:             256
Current owner:            SP A
Auto-trespass:            DISABLED
Auto-assign:              DISABLED
Write cache:              ENABLED
Read cache:               ENABLED
Idle Threshold:           0
Idle Delay Time:          20
Write Aside Size:         1023
Default Owner:            SP A
Rebuild Priority:         Low
Verify Priority:           Low
Prct Reads Forced Flushed: 0
Prct Writes Forced Flushed: 0
Prct Rebuilt:             100
Prct Bound:               100
Lun Capacity:             202357

```

```

Enclosure 0 Disk 0 Enabled
Enclosure 0 Disk 4 Enabled
Enclosure 0 Disk 8 Enabled
Enclosure 0 Disk 1 Enabled
Enclosure 0 Disk 5 Enabled
Enclosure 0 Disk 9 Enabled
Enclosure 0 Disk 2 Enabled
Enclosure 0 Disk 6 Enabled
Enclosure 1 Disk 0 Enabled
Enclosure 0 Disk 3 Enabled
Enclosure 0 Disk 7 Enabled
Enclosure 1 Disk 1 Enabled

```

Statistics logging is disabled.
 Certain fields are not printed if statistics logging is not enabled.

```

C:\Program Files\UNISYS\Navisphere CLI>navicli -d \\.\SCSI2:0:2:1 getlun
1
Prefetch size (blocks) = 4
Prefetch multiplier = 4

```

```

Segment size (blocks) = 4
Segment multiplier = 4
Maximum prefetch (blocks) = 512
Prefetch Disable Size (blocks) = 129
Prefetch idle count = 40

```

```

Variable length prefetching YES
Prefetched data retained YES

```

Read cache configured according to specified parameters.

Minimum latency reads N/A

```

RAID Type:                RAID1/0
RAIDGroup ID:             1
State:                    Bound
Stripe Crossing:          0
Element Size:             256
Current owner:            SP B
Auto-trespass:            DISABLED
Auto-assign:              DISABLED
Write cache:              ENABLED
Read cache:               ENABLED
Idle Threshold:           0
Idle Delay Time:          20
Write Aside Size:         1023
Default Owner:            SP A
Rebuild Priority:         Low
Verify Priority:           Low
Prct Reads Forced Flushed: 0
Prct Writes Forced Flushed: 0
Prct Rebuilt:             100
Prct Bound:               100
Lun Capacity:             202357

```

```

Enclosure 2 Disk 0 Enabled
Enclosure 2 Disk 4 Enabled
Enclosure 2 Disk 8 Enabled
Enclosure 2 Disk 1 Enabled
Enclosure 2 Disk 5 Enabled
Enclosure 2 Disk 9 Enabled
Enclosure 2 Disk 2 Enabled
Enclosure 2 Disk 6 Enabled
Enclosure 3 Disk 0 Enabled
Enclosure 2 Disk 3 Enabled
Enclosure 2 Disk 7 Enabled
Enclosure 3 Disk 1 Enabled

```

Statistics logging is disabled.
Certain fields are not printed if statistics
logging is not enabled.

Windows Datacenter Server Configuration Information

System Information report written at: 7/19/2001 8:09:59 PM
System Name: MV-LE030P1
[System Summary]

Item	Value
OS Name	Microsoft Windows XP Datacenter Server
Version	5.1.2509 Build 2509
OS Manufacturer	Microsoft Corporation
System Name	MV-LE030P1
System Manufacturer	UNISYS
System Model	Horizon
System Type	X86-based PC
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~899 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~899 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel ~900 Mhz

Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~899 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz
Processor	x86 Family 6 Model 10 Stepping 4 GenuineIntel	~900 Mhz

BIOS Version/Date	Phoenix Technologies LTD BIOS Release 000048 Build 000328, 6/20/2001	
SMBIOS Version	2.31	
Windows Directory	C:\WINDOWS	
System Directory	C:\WINDOWS\System32	
Boot Device	\Device\HarddiskVolume1	
Locale	United States	
Hardware Abstraction Layer	Version = "5.1.2473.124 (Lab01_N(bradw).010507-1232)"	
User Name	MV-LE030P1\Administrator	
Time Zone	Pacific Daylight Time	
Total Physical Memory	65,536.00 MB	
Available Physical Memory	62.98 GB	
Total Virtual Memory	128.67 GB	
Available Virtual Memory	127.89 GB	
Page File Space	65.05 GB	
Page File	C:\pagefile.sys	

[Hardware Resources]

[Conflicts/Sharing]

Resource	Device	
I/O Port 0x00000000-0x0000102F		PCI bus
I/O Port 0x00000000-0x0000102F		Direct memory access controller

Memory Address 0xEF800000-0xEFFFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xEF800000-0xEFFFFFFF Unisys Fibre Channel Driver v1.3.15

Memory Address 0xF8800000-0xF8FFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xF8800000-0xF8FFFFFF Unisys Fibre Channel Driver v1.3.15

Memory Address 0xF3000000-0xF37FFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xF3000000-0xF37FFFFFF Unisys Fibre Channel Driver v1.3.15

Memory Address 0xF0800000-0xF0FFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xF0800000-0xF0FFFFFF Unisys Fibre Channel Driver v1.3.15

I/O Port 0x000003C0-0x000003DF DEC 21152 PCI to PCI bridge

I/O Port 0x000003C0-0x000003DF Console Manager Video Driver

IRQ 8 System CMOS/real time clock

IRQ 8 Adaptec SCSI Card 39160 - Ultra160 SCSI

Memory Address 0xFB000000-0xFB7FFFFFF PCI bus

Memory Address 0xFB000000-0xFB7FFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xFB000000-0xFB7FFFFFF Unisys Fibre Channel Driver v1.3.15

Memory Address 0xFD000000-0xFDFFFFFF PCI bus

Memory Address 0xFD000000-0xFDFFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xFD000000-0xFDFFFFFF Unisys Fibre Channel Driver v1.3.15

Memory Address 0xF9100000-0xFAFFFFFF PCI bus

Memory Address 0xF9100000-0xFAFFFFFF cLAN Host Adapter #4

Memory Address 0xF6000000-0xF67FFFFFF PCI bus

Memory Address 0xF6000000-0xF67FFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xF6000000-0xF67FFFFFF Unisys Fibre Channel Driver v1.3.15

Memory Address 0xEF000000-0xEF3FFFFFF PCI bus

Memory Address 0xEF000000-0xEF3FFFFFF Intel(R) PRO/100+ PCI Adapter

Memory Address 0xEE800000-0xEEFFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xEE800000-0xEEFFFFFF Unisys Fibre Channel Driver v1.3.15

Memory Address 0xA0000-0xBFFFF PCI bus

Memory Address 0xA0000-0xBFFFF DEC 21152 PCI to PCI bridge

Memory Address 0xA0000-0xBFFFF Console Manager Video Driver

Memory Address 0xF4100000-0xF5FFFFFF PCI bus

Memory Address 0xF4100000-0xF5FFFFFF cLAN Host Adapter #3

Memory Address 0xF3800000-0xF3FFFFFF PCI bus

Memory Address 0xF3800000-0xF3FFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xF3800000-0xF3FFFFFF Unisys Fibre Channel Driver v1.3.15

I/O Port 0x000003B0-0x000003BB DEC 21152 PCI to PCI bridge

I/O Port 0x000003B0-0x000003BB Console Manager Video Driver

I/O Port 0x00001040-0x000013FF PCI bus

I/O Port 0x00001040-0x000013FF Intel(R) PRO/100+ PCI Adapter

IRQ 0 System timer

IRQ 0 Intel(r) 82371AB/EB PCI to USB Universal Host Controller

Memory Address 0xE8100000-0xEDFFFFFF DEC 21152 PCI to PCI bridge

Memory Address 0xE8100000-0xEDFFFFFF Console Manager Video Driver

[DMA]

Resource	Device	Status
Channel 4	Direct memory access controller	OK
Channel 1	ECP Printer Port (LPT1)	OK
Channel 2	Standard floppy disk controller	OK

[Forced Hardware]

Device PNP Device ID

[I/O]

Resource	Device	Status
0x00001400-0x0000140F	PCI bus	OK
0x00001410-0x0000141F	PCI bus	OK
0x00001420-0x000017FF	PCI bus	OK
0x00000000-0x0000102F	PCI bus	OK
0x00000000-0x0000102F	Direct memory access controller	OK
0x00000A79-0x00000A79	ISAPNP Read Data Port	OK
0x00000279-0x00000279	ISAPNP Read Data Port	OK
0x00000274-0x00000277	ISAPNP Read Data Port	OK
0x00000081-0x0000008F	Direct memory access controller	OK
0x000000C0-0x000000DF	Direct memory access controller	OK
0x00000020-0x00000021	Programmable interrupt controller	OK
0x000000A0-0x000000A1	Programmable interrupt controller	OK
0x00000070-0x00000071	System CMOS/real time clock	OK
0x00000040-0x00000043	System timer	OK
0x00000061-0x00000061	System speaker	OK
0x00000101-0x0000016F	System speaker	OK
0x00000178-0x000001CD	System speaker	OK
0x000001D0-0x000001EF	System speaker	OK
0x000001F8-0x00000273	System speaker	OK
0x00000280-0x000002E7	System speaker	OK
0x000002F0-0x000002F7	System speaker	OK
0x00000300-0x00000377	System speaker	OK
0x00000380-0x000003AF	System speaker	OK
0x000003BC-0x000003BF	System speaker	OK
0x000003E0-0x000003E7	System speaker	OK
0x000003F6-0x000003F6	System speaker	OK
0x00000501-0x0000056F	System speaker	OK
0x00000578-0x000005CD	System speaker	OK
0x000005D0-0x000005EF	System speaker	OK
0x000005F8-0x00000677	System speaker	OK
0x00000680-0x000006E7	System speaker	OK
0x000006F0-0x000006F7	System speaker	OK
0x00000700-0x00000777	System speaker	OK

0x00000780-0x000007AF System speaker OK
 0x000007BC-0x000007BF System speaker OK
 0x000007E0-0x000007E7 System speaker OK
 0x000007F6-0x000007F6 System speaker OK
 0x00000901-0x0000096F System speaker OK
 0x00000978-0x000009CD System speaker OK
 0x000009D0-0x000009EF System speaker OK
 0x000009F8-0x00000A77 System speaker OK
 0x00000A80-0x00000AE7 System speaker OK
 0x00000AF0-0x00000AF7 System speaker OK
 0x00000B00-0x00000B77 System speaker OK
 0x00000B80-0x00000BAF System speaker OK
 0x00000BBC-0x00000BBF System speaker OK
 0x00000BE0-0x00000BE7 System speaker OK
 0x00000BF6-0x00000BF6 System speaker OK
 0x00000D01-0x00000D6F System speaker OK
 0x00000D78-0x00000DCD System speaker OK
 0x00000DD0-0x00000DEF System speaker OK
 0x00000DF8-0x00000E77 System speaker OK
 0x00000E80-0x00000EE7 System speaker OK
 0x00000EF0-0x00000EF7 System speaker OK
 0x00000F00-0x00000F77 System speaker OK
 0x00000F80-0x00000FAF System speaker OK
 0x00000FBC-0x00000FBF System speaker OK
 0x00000FE0-0x00000FE7 System speaker OK
 0x00000FF6-0x00000FF6 System speaker OK
 0x00001101-0x0000116F System speaker OK
 0x00001178-0x000011CD System speaker OK
 0x000011D0-0x000011EF System speaker OK
 0x000011F8-0x00001277 System speaker OK
 0x00001280-0x000012E7 System speaker OK
 0x000012F0-0x000012F7 System speaker OK
 0x00001300-0x00001377 System speaker OK
 0x00001380-0x000013AF System speaker OK
 0x000013BC-0x000013BF System speaker OK
 0x000013E0-0x000013E7 System speaker OK
 0x000013F6-0x000013F6 System speaker OK
 0x00001501-0x0000156F System speaker OK
 0x00001578-0x000015CD System speaker OK
 0x000015D0-0x000015EF System speaker OK
 0x000015F8-0x00001677 System speaker OK
 0x00001680-0x000016E7 System speaker OK
 0x000016F0-0x000016F7 System speaker OK
 0x00001700-0x00001777 System speaker OK
 0x00001780-0x000017AF System speaker OK
 0x000017BC-0x000017BF System speaker OK
 0x000017E0-0x000017E7 System speaker OK
 0x000017F6-0x000017F6 System speaker OK
 0x00001901-0x0000196F System speaker OK
 0x00001978-0x000019CD System speaker OK
 0x000019D0-0x000019EF System speaker OK
 0x000019F8-0x00001A77 System speaker OK
 0x00001A80-0x00001AE7 System speaker OK
 0x00001AF0-0x00001AF7 System speaker OK
 0x00001B00-0x00001B77 System speaker OK
 0x00001B80-0x00001BAF System speaker OK
 0x00001BBC-0x00001BBF System speaker OK
 0x00001BE0-0x00001BE7 System speaker OK
 0x00001BF6-0x00001BF6 System speaker OK

0x00001D01-0x00001D6F System speaker OK
 0x00001D78-0x00001DCD System speaker OK
 0x00001DD0-0x00001DEF System speaker OK
 0x00001DF8-0x00001E77 System speaker OK
 0x00001E80-0x00001EE7 System speaker OK
 0x00001EF0-0x00001EF7 System speaker OK
 0x00001F00-0x00001F77 System speaker OK
 0x00001F80-0x00001FAF System speaker OK
 0x00001FBC-0x00001FBF System speaker OK
 0x00001FE0-0x00001FE7 System speaker OK
 0x00001FF6-0x00001FF6 System speaker OK
 0x000002F8-0x000002FF Communications Port (COM2) OK
 0x00000378-0x0000037F ECP Printer Port (LPT1) OK
 0x00000778-0x0000077F ECP Printer Port (LPT1) OK
 0x000003F0-0x000003F5 Standard floppy disk controller OK
 0x000003F7-0x000003F7 Standard floppy disk controller OK
 0x00000060-0x00000060 Standard 101/102-Key or Microsoft Natural PS/2 Keyboard OK
 0x00000064-0x00000064 Standard 101/102-Key or Microsoft Natural PS/2 Keyboard OK
 0x00001020-0x0000102F Intel(r) 82371AB/EB PCI Bus Master IDE Controller OK
 0x00001000-0x0000101F Intel(r) 82371AB/EB PCI to USB Universal Host Controller OK
 0x000003B0-0x000003BB DEC 21152 PCI to PCI bridge OK
 0x000003B0-0x000003BB Console Manager Video Driver OK
 0x000003C0-0x000003DF DEC 21152 PCI to PCI bridge OK
 0x000003C0-0x000003DF Console Manager Video Driver OK
 0x00001030-0x0000103F PCI bus OK
 0x00001040-0x000013FF PCI bus OK
 0x00001040-0x000013FF Intel(R) PRO/100+ PCI Adapter OK
 0x00001100-0x000011FF Adaptec SCSI Card 39160 - Ultra160 SCSI OK
 0x00001200-0x000012FF Adaptec SCSI Card 39160 - Ultra160 SCSI OK
 0x00001800-0x0000180F PCI bus OK
 0x00001810-0x0000181F PCI bus OK
 0x00001820-0x00001BFF PCI bus OK
 0x00001C00-0x00001C0F PCI bus OK
 0x00001C10-0x00001C1F PCI bus OK
 0x00001C20-0x00001FFF PCI bus OK
 0x00002000-0x0000200F PCI bus OK
 0x00002010-0x0000201F PCI bus OK
 0x00002020-0x000023FF PCI bus OK
 0x00002400-0x0000240F PCI bus OK
 0x00002410-0x0000241F PCI bus OK
 0x00002420-0x000027FF PCI bus OK
 0x00002800-0x0000280F PCI bus OK
 0x00002810-0x0000281F PCI bus OK
 0x00002820-0x00002BFF PCI bus OK
 0x00002C00-0x00002C0F PCI bus OK
 0x00002C10-0x00002C1F PCI bus OK
 0x00002C20-0x0000FFFF PCI bus OK

[IRQs]

Resource	Device	Status
IRQ 23	Microsoft ACPI-Compliant System	OK
IRQ 26	Unisys Fibre Channel Driver v1.3.15	OK

```

IRQ 8 System CMOS/real time clock OK
IRQ 8 Adaptec SCSI Card 39160 - Ultra160 SCSI OK
IRQ 0 System timer OK
IRQ 0 Intel(r) 82371AB/EB PCI to USB Universal Host Controller OK

IRQ 3 Communications Port (COM2) OK
IRQ 6 Standard floppy disk controller OK
IRQ 12 PS/2 Compatible Mouse OK
IRQ 1 Standard 101/102-Key or Microsoft Natural PS/2 Keyboard OK

IRQ 2 Console Manager control device OK
IRQ 5 Unisys Fibre Channel Driver v1.3.15 OK
IRQ 9 Adaptec SCSI Card 39160 - Ultra160 SCSI OK
IRQ 10 Intel(R) PRO/100+ PCI Adapter OK
IRQ 49 Unisys Fibre Channel Driver v1.3.15 OK
IRQ 81 Unisys Fibre Channel Driver v1.3.15 OK
IRQ 96 Unisys Fibre Channel Driver v1.3.15 OK
IRQ 105 cLAN Host Adapter #3 OK
IRQ 121 Unisys Fibre Channel Driver v1.3.15 OK
IRQ 144 Unisys Fibre Channel Driver v1.3.15 OK
IRQ 153 cLAN Host Adapter #4 OK
IRQ 168 Unisys Fibre Channel Driver v1.3.15 OK
IRQ 177 Unisys Fibre Channel Driver v1.3.15 OK

```

[Memory]

Resource	Device	Status
0xEF400000-0xEFFFFFFF	PCI bus	OK
0xEF800000-0xEFFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xEF800000-0xEFFFFFFF	Unisys Fibre Channel Driver v1.3.15	OK
0xF0000000-0xF00FFFFFFF	PCI bus	OK
0xF0100000-0xF03FFFFFFF	PCI bus	OK
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	DEC 21152 PCI to PCI bridge	OK
0xA0000-0xBFFFF	Console Manager Video Driver	OK
0xC0000-0xDFFFF	PCI bus	OK
0xE0000-0xFFFFF	PCI bus	OK
0xE8000000-0xEE1FFFFFFF	PCI bus	OK
0xE8100000-0xEDFFFFFFF	DEC 21152 PCI to PCI bridge	OK
0xE8100000-0xEDFFFFFFF	Console Manager Video Driver	OK
0xEA000000-0xEBFFFFFFF	Console Manager control device	OK
0xEC000000-0xEDFFFFFFF	Console Manager Video Driver	OK
0xEE200000-0xEFFFFFFF	PCI bus	OK
0xEE800000-0xEFFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xEE800000-0xEFFFFFFF	Unisys Fibre Channel Driver v1.3.15	OK
0xEF000000-0xEF3FFFFFFF	PCI bus	OK
0xEF000000-0xEF3FFFFFFF	Intel(R) PRO/100+ PCI Adapter	OK
0xEF100000-0xEF100FFF	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
0xEF101000-0xEF101FFF	Adaptec SCSI Card 39160 - Ultra160 SCSI	OK
0xEF102000-0xEF102FFF	Intel(R) PRO/100+ PCI Adapter	OK
0xF0400000-0xF0FFFFFFF	PCI bus	OK
0xF0800000-0xF0FFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xF0800000-0xF0FFFFFFF	Unisys Fibre Channel Driver v1.3.15	OK
0xF1000000-0xF10FFFFFFF	PCI bus	OK
0xF1100000-0xF13FFFFFFF	PCI bus	OK
0xF1400000-0xF14FFFFFFF	PCI bus	OK
0xF1500000-0xF15FFFFFFF	PCI bus	OK

```

0xF1600000-0xF37FFFFFFF PCI bus OK
0xF3000000-0xF37FFFFFFF DEC 21154 PCI to PCI bridge OK
0xF3000000-0xF37FFFFFFF Unisys Fibre Channel Driver v1.3.15 OK
0xF3800000-0xF3FFFFFFF PCI bus OK
0xF3800000-0xF3FFFFFFF DEC 21154 PCI to PCI bridge OK
0xF3800000-0xF3FFFFFFF Unisys Fibre Channel Driver v1.3.15 OK
0xF4000000-0xF40FFFFFFF PCI bus OK
0xF4100000-0xF5FFFFFFF PCI bus OK
0xF4100000-0xF5FFFFFFF cLAN Host Adapter #3 OK
0xF4120000-0xF413FFFFF cLAN Host Adapter #3 OK
0xF4200000-0xF43FFFFFFF cLAN Host Adapter #3 OK
0xF5000000-0xF5FFFFFFF cLAN Host Adapter #3 OK
0xF6000000-0xF67FFFFFFF PCI bus OK
0xF6000000-0xF67FFFFFFF DEC 21154 PCI to PCI bridge OK
0xF6000000-0xF67FFFFFFF Unisys Fibre Channel Driver v1.3.15 OK
0xF6800000-0xF7FFFFFFF PCI bus OK
0xF8000000-0xF83FFFFFFF PCI bus OK
0xF8400000-0xF8FFFFFFF PCI bus OK
0xF8800000-0xF8FFFFFFF DEC 21154 PCI to PCI bridge OK
0xF8800000-0xF8FFFFFFF Unisys Fibre Channel Driver v1.3.15 OK
0xF9000000-0xF90FFFFFFF PCI bus OK
0xF9100000-0xFAFFFFFFF PCI bus OK
0xF9100000-0xFAFFFFFFF cLAN Host Adapter #4 OK
0xF9120000-0xF913FFFFF cLAN Host Adapter #4 OK
0xF9200000-0xF93FFFFFFF cLAN Host Adapter #4 OK
0xFA000000-0xFAFFFFFFF cLAN Host Adapter #4 OK
0xFB000000-0xFB7FFFFFFF PCI bus OK
0xFB000000-0xFB7FFFFFFF DEC 21154 PCI to PCI bridge OK
0xFB000000-0xFB7FFFFFFF Unisys Fibre Channel Driver v1.3.15 OK
0xFB800000-0xFCFFFFFFF PCI bus OK
0xFD000000-0xFDFFFFFFF PCI bus OK
0xFD000000-0xFDFFFFFFF DEC 21154 PCI to PCI bridge OK
0xFD000000-0xFDFFFFFFF Unisys Fibre Channel Driver v1.3.15 OK

```

[Components]

[Multimedia]

[Audio Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size
		Creation Date				
		c:\windows\system32\msg723.acm	Microsoft Corporation			OK
		C:\WINDOWS\System32\MSG723.ACM		4.4.3385		116.00 KB
		(118,784 bytes)		6/20/2001 12:53 PM		
		c:\windows\system32\tsssoft32.acm	DSP GROUP, INC.			OK
		C:\WINDOWS\System32\TSSOFT32.ACM		1.01	8.00 KB	(8,192 bytes)
		6/16/2001 8:42 AM				
		c:\windows\system32\iac25_32.ax	Intel Corporation			Indeo® audio
		software	OK	C:\WINDOWS\System32\IAC25_32.AX		2.05.53 195.00
		KB (199,680 bytes)		6/16/2001 8:46 AM		
		c:\windows\system32\msadp32.acm	Microsoft Corporation			OK
		C:\WINDOWS\System32\MSADP32.ACM		5.1.2496.0 (main.010615-1711)		
		13.00 KB (13,312 bytes)		6/16/2001 8:39 AM		

```

c:\windows\system32\imaadp32.acm      Microsoft Corporation      OK
C:\WINDOWS\System32\IMAADP32.ACM    5.1.2496.0 (main.010615-1711)
14.50 KB (14,848 bytes)             6/16/2001 8:42 AM
c:\windows\system32\msg711.acm      Microsoft Corporation      OK
C:\WINDOWS\System32\MSG711.ACM     5.1.2496.0 (main.010615-1711)
9.00 KB (9,216 bytes)                6/16/2001 8:42 AM
c:\windows\system32\msgsm32.acm     Microsoft Corporation      OK
C:\WINDOWS\System32\MSGSM32.ACM    5.1.2496.0 (main.010615-1711)
19.50 KB (19,968 bytes)             6/16/2001 8:42 AM

```

[Video Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size
		Creation Date				
c:\windows\system32\msvidc32.dll	Microsoft Corporation		OK			
		C:\WINDOWS\System32\MSVIDC32.DLL		5.1.2496.0 (main.010615-1711)		25.00 KB (25,600 bytes)
				6/16/2001 8:44 AM		
c:\windows\system32\iyuv_32.dll	Microsoft Corporation		OK			
		C:\WINDOWS\System32\IYUV_32.DLL		5.1.2496.0 (main.010615-1711)		44.50 KB (45,568 bytes)
				6/16/2001 1:43 AM		
c:\windows\system32\ir41_32.ax	Intel Corporation		OK			
		C:\WINDOWS\System32\IR41_32.AX		4.51.16.03		828.50 KB
				6/16/2001 8:46 AM		(848,384 bytes)
c:\windows\system32\msrle32.dll	Microsoft Corporation		OK			
		C:\WINDOWS\System32\MSRLE32.DLL		5.1.2496.0 (main.010615-1711)		9.50 KB (9,728 bytes)
				6/16/2001 8:44 AM		
c:\windows\system32\msh261.drv	Microsoft Corporation		OK			
		C:\WINDOWS\System32\MSH261.DRV		4.4.3385		180.00 KB
				6/20/2001 12:53 PM		(184,320 bytes)
c:\windows\system32\ir32_32.dll	Intel(R) Corporation		OK			
		C:\WINDOWS\System32\IR32_32.DLL		Not Available		194.50 KB
				6/16/2001 8:43 AM		(199,168 bytes)
c:\windows\system32\msyuv.dll	Microsoft Corporation		OK			
		C:\WINDOWS\System32\MSYUV.DLL		5.1.2496.0 (main.010615-1711)		16.00 KB (16,384 bytes)
				6/16/2001 1:44 AM		
c:\windows\system32\tsbyuv.dll	Microsoft Corporation		OK			
		C:\WINDOWS\System32\TSBYUV.DLL		5.1.2496.0 (main.010615-1711)		8.00 KB (8,192 bytes)
				6/16/2001 1:45 AM		
c:\windows\system32\msh263.drv	Microsoft Corporation		OK			
		C:\WINDOWS\System32\MSH263.DRV		4.4.3385		280.00 KB
				6/16/2001 1:46 AM		(286,720 bytes)
c:\windows\system32\ir50_32.dll	Intel Corporation	Indeo® video				
		C:\WINDOWS\System32\IR50_32.DLL		R.5.10.15.2.55		737.50 KB (755,200 bytes)
				6/16/2001 8:43 AM		
c:\windows\system32\iccvid.dll	Radius Inc.		OK			
		C:\WINDOWS\System32\ICCVID.DLL		1.10.0.6		108.00 KB
				6/16/2001 8:43 AM		(110,592 bytes)

[CD-ROM]

Item Value

[Sound Device]

Item Value

[Display]

Item Value

```

Name      Console Manager Video Driver
PNP Device ID
          PCI\VEN_1013&DEV_00D6&SUBSYS_80D61018&REV_03\4&3A6BE3A2&0&1010
Adapter Type      Cirrus Logic 5465, Unisys Corporation compatible
Adapter Description      Console Manager Video Driver
Adapter RAM      2.00 MB (2,097,152 bytes)
Installed Drivers      unesconf.dll
Driver Version      6,0,2,9
INF File      oem1.inf (unescons section)
Color Planes      1
Color Table Entries      256
Resolution      800 x 600 x 72 hertz
Bits/Pixel      8
Memory Address      0xEC000000-0xEDFFFFFF
Memory Address      0xE8100000-0xEDFFFFFF
I/O Port      0x000003B0-0x000003BB
I/O Port      0x000003C0-0x000003DF
Memory Address      0xA0000-0xBFFFF
Driver c:\windows\system32\drivers\unescons.sys (6,0,2,9, 249.73 KB
(255,719 bytes), 5/3/2001 1:01 PM)

[Infrared]
Item      Value

[Input]

[Keyboard]
Item      Value
Description      Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Name      Enhanced (101- or 102-key)
Layout      00000409
PNP Device ID      ACPI\PNP0303\2
Number of Function Keys      12
I/O Port      0x00000060-0x00000060
I/O Port      0x00000064-0x00000064
IRQ Channel      IRQ 1
Driver c:\windows\system32\drivers\i8042prt.sys (5.1.2496.0 (main.010615-
1711), 49.63 KB (50,816 bytes), 6/16/2001 8:28 AM)

[Pointing Device]
Item      Value
Hardware Type      PS/2 Compatible Mouse
Number of Buttons      3
Status      OK
PNP Device ID      ACPI\PNP0F13\2
Power Management Supported      No
Double Click Threshold      6
Handedness      Right Handed Operation
IRQ Channel      IRQ 12
Driver c:\windows\system32\drivers\i8042prt.sys (5.1.2496.0 (main.010615-
1711), 49.63 KB (50,816 bytes), 6/16/2001 8:28 AM)

```

[Modem]

Item Value

[Network]

[Adapter]

Item Value

Name [00000001] Intel(R) PRO/100+ Server Adapter (PILA8470B)
 Adapter Type Not Available
 Product Type Intel(R) PRO/100+ Server Adapter (PILA8470B)
 Installed Yes
 PNP Device ID Not Available
 Last Reset 7/18/2001 12:22 PM
 Index 1
 Service Name E100B
 IP Address 192.59.13.213
 IP Subnet 255.255.255.0
 Default IP Gateway 192.59.13.250
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:A0:C9:D9:76:D5

Name [00000002] RAS Async Adapter
 Adapter Type Not Available
 Product Type RAS Async Adapter
 Installed Yes
 PNP Device ID Not Available
 Last Reset 7/18/2001 12:22 PM
 Index 2
 Service Name AsyncMac
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address Not Available

Name [00000003] WAN Miniport (L2TP)
 Adapter Type Not Available
 Product Type WAN Miniport (L2TP)
 Installed Yes
 PNP Device ID ROOT\MS_L2TPMINIPORT\0000
 Last Reset 7/18/2001 12:22 PM
 Index 3
 Service Name Rasl2tp
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available

DHCP Lease Obtained Not Available
 MAC Address Not Available
 Driver c:\windows\system32\drivers\rasl2tp.sys (5.1.2496.0 (main.010615-1711), 46.88 KB (48,000 bytes), 6/16/2001 8:28 AM)

Name [00000004] WAN Miniport (PPTP)
 Adapter Type Wide Area Network (WAN)
 Product Type WAN Miniport (PPTP)
 Installed Yes
 PNP Device ID ROOT\MS_PPTPMINIPORT\0000
 Last Reset 7/18/2001 12:22 PM
 Index 4
 Service Name PptpMiniport
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 50:50:54:50:30:30
 Driver c:\windows\system32\drivers\raspptp.sys (5.1.2496.0 (main.010615-1711), 45.50 KB (46,592 bytes), 6/16/2001 8:28 AM)

Name [00000005] WAN Miniport (PPPOE)
 Adapter Type Wide Area Network (WAN)
 Product Type WAN Miniport (PPPOE)
 Installed Yes
 PNP Device ID ROOT\MS_PPPOEMINIPORT\0000
 Last Reset 7/18/2001 12:22 PM
 Index 5
 Service Name RasPppoe
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 33:50:6F:45:30:30
 Driver c:\windows\system32\drivers\raspppoe.sys (5.1.2496.0 (main.010615-1711), 37.88 KB (38,784 bytes), 6/15/2001 7:30 PM)

Name [00000006] Direct Parallel
 Adapter Type Not Available
 Product Type Direct Parallel
 Installed Yes
 PNP Device ID ROOT\MS_PTMINIPORT\0000
 Last Reset 7/18/2001 12:22 PM
 Index 6
 Service Name Raspti
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled No
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address Not Available

Driver c:\windows\system32\drivers\raspti.sys (5.1.2496.0 (main.010615-1711), 16.13 KB (16,512 bytes), 6/15/2001 7:30 PM)

Name [00000007] WAN Miniport (IP)
Adapter Type Not Available
Product Type WAN Miniport (IP)
Installed Yes
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 7/18/2001 12:22 PM
Index 7
Service Name NdisWan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available

Driver c:\windows\system32\drivers\ndiswan.sys (5.1.2496.0 (main.010615-1711), 85.00 KB (87,040 bytes), 6/16/2001 8:28 AM)

Name [00000008] Intel(R) PRO/100+ PCI Adapter
Adapter Type Ethernet 802.3
Product Type Intel(R) PRO/100+ PCI Adapter
Installed Yes
PNP Device ID
PCI\VEN_8086&DEV_1229&SUBSYS_00098086&REV_05\3&20FEA912&0&10

Last Reset 7/18/2001 12:22 PM
Index 8
Service Name E100B
IP Address 192.59.13.213
IP Subnet 255.255.255.0

Default IP Gateway 192.59.13.250
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:A0:C9:D9:76:D5
Memory Address 0xEF102000-0xEF102FFF
I/O Port 0x00001040-0x000013FF
Memory Address 0xEF000000-0xEF3FFFFF
IRQ Channel IRQ 10

Driver c:\windows\system32\drivers\e100b325.sys (5.41.22.0000 built by: WinDDK, 115.00 KB (117,760 bytes), 6/20/2001 5:36 AM)

Name [00000009] cLAN Host Adapter
Adapter Type Not Available
Product Type cLAN Host Adapter
Installed Yes
PNP Device ID
PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&146CA173&0&18

Last Reset 7/18/2001 12:22 PM
Index 9
Service Name GNINDIS
IP Address Not Available
IP Subnet Not Available

Default IP Gateway Not Available

DHCP Enabled Yes
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Driver c:\windows\system32\drivers\gnindis.sys (4.1.1, 22.07 KB (22,598 bytes), 6/20/2001 3:35 PM)

Name [00000010] cLAN Host Adapter
Adapter Type Not Available
Product Type cLAN Host Adapter
Installed Yes
PNP Device ID
PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&23FF515D&0&08

Last Reset 7/18/2001 12:22 PM
Index 10
Service Name GNINDIS
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available

DHCP Enabled Yes
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Driver c:\windows\system32\drivers\gnindis.sys (4.1.1, 22.07 KB (22,598 bytes), 6/20/2001 3:35 PM)

Name [00000011] cLAN Host Adapter
Adapter Type Ethernet 802.3
Product Type cLAN Host Adapter
Installed Yes
PNP Device ID
PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&A985F74&0&08

Last Reset 7/18/2001 12:22 PM
Index 11
Service Name GNINDIS
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available

DHCP Enabled Yes
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:90:FA:00:14:BC
Memory Address 0xF4120000-0xF413FFFF
Memory Address 0xF4200000-0xF43FFFFF
Memory Address 0xF5000000-0xF5FFFFFF
Memory Address 0xF4100000-0xF5FFFFFF
IRQ Channel IRQ 105
Driver c:\windows\system32\drivers\gnindis.sys (4.1.1, 22.07 KB (22,598 bytes), 6/20/2001 3:35 PM)

Name [00000012] cLAN Host Adapter
Adapter Type Ethernet 802.3
Product Type cLAN Host Adapter
Installed Yes

PNP Device ID PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&3BCEF44&0&08

Last Reset 7/18/2001 12:22 PM
 Index 12
 Service Name GNINDIS
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled Yes
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address 00:90:FA:00:1C:45
 Memory Address 0xF9120000-0xF913FFFF
 Memory Address 0xF9200000-0xF93FFFFF
 Memory Address 0xF9100000-0xFAFFFFFFF
 IRQ Channel IRQ 153
 Driver c:\windows\system32\drivers\gnindis.sys (4.1.1, 22.07 KB (22,598 bytes), 6/20/2001 3:35 PM)

Name [00000013] cLAN Host Adapter
 Adapter Type Not Available
 Product Type cLAN Host Adapter
 Installed Yes
 PNP Device ID
 PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&300BC0BE&0&08

Last Reset 7/18/2001 12:22 PM
 Index 13
 Service Name GNINDIS
 IP Address Not Available
 IP Subnet Not Available
 Default IP Gateway Not Available
 DHCP Enabled Yes
 DHCP Server Not Available
 DHCP Lease Expires Not Available
 DHCP Lease Obtained Not Available
 MAC Address Not Available
 Driver c:\windows\system32\drivers\gnindis.sys (4.1.1, 22.07 KB (22,598 bytes), 6/20/2001 3:35 PM)

[Protocol]

Item	Value
Name	MSAFD Tcpip [TCP/IP]
Connectionless Service	No
Guarantees Delivery	Yes
Guarantees Sequencing	Yes
Maximum Address Size	16 bytes
Maximum Message Size	0 bytes
Message Oriented	No
Minimum Address Size	16 bytes
Pseudo Stream Oriented	No
Supports Broadcasting	No
Supports Connect Data	No
Supports Disconnect Data	No
Supports Encryption	No
Supports Expedited Data	Yes
Supports Graceful Closing	Yes

Supports Guaranteed Bandwidth No
 Supports Multicasting No

Name MSAFD Tcpip [UDP/IP]
 Connectionless Service Yes
 Guarantees Delivery No
 Guarantees Sequencing No
 Maximum Address Size 16 bytes
 Maximum Message Size 63.93 KB (65,467 bytes)
 Message Oriented Yes
 Minimum Address Size 16 bytes
 Pseudo Stream Oriented No
 Supports Broadcasting Yes
 Supports Connect Data No
 Supports Disconnect Data No
 Supports Encryption No
 Supports Expedited Data No
 Supports Graceful Closing No
 Supports Guaranteed Bandwidth No
 Supports Multicasting Yes

Name RSVP UDP Service Provider
 Connectionless Service Yes
 Guarantees Delivery No
 Guarantees Sequencing No
 Maximum Address Size 16 bytes
 Maximum Message Size 63.93 KB (65,467 bytes)
 Message Oriented Yes
 Minimum Address Size 16 bytes
 Pseudo Stream Oriented No
 Supports Broadcasting Yes
 Supports Connect Data No
 Supports Disconnect Data No
 Supports Encryption Yes
 Supports Expedited Data No
 Supports Graceful Closing No
 Supports Guaranteed Bandwidth No
 Supports Multicasting Yes

Name RSVP TCP Service Provider
 Connectionless Service No
 Guarantees Delivery Yes
 Guarantees Sequencing Yes
 Maximum Address Size 16 bytes
 Maximum Message Size 0 bytes
 Message Oriented No
 Minimum Address Size 16 bytes
 Pseudo Stream Oriented No
 Supports Broadcasting No
 Supports Connect Data No
 Supports Disconnect Data No
 Supports Encryption Yes
 Supports Expedited Data Yes
 Supports Graceful Closing Yes
 Supports Guaranteed Bandwidth No
 Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C64D1153-FBA0-4B95-91B3-C9F10EBEE86D}] SEQPACKE 8

Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C64D1153-FBA0-4B95-91B3-C9F10EBEE86D}] DATAGRAM 8
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{87DA539C-4C8C-41D8-861C-11BD6DC1EBD9}] SEQPACKET 7
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{87DA539C-4C8C-41D8-861C-11BD6DC1EBD9}] DATAGRAM 7
Connectionless Service Yes
Guarantees Delivery No

Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C52D958A-FCD8-419F-8F1F-C49AD99857A8}] SEQPACKET 6
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C52D958A-FCD8-419F-8F1F-C49AD99857A8}] DATAGRAM 6
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{7AABDE4C-6516-4E60-B5B9-7CE9B249F7E2}] SEQPACKET 5
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes

Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{7AABDE4C-6516-4E60-B5B9-7CE9B249F7E2}] DATAGRAM 5
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{D28D75FD-678D-4CFA-A628-1EFE92827412}] SEQPACKET 4
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{D28D75FD-678D-4CFA-A628-1EFE92827412}] DATAGRAM 4
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes

Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{6E27CD90-8318-46C1-9520-F09DB7A90D0F}] SEQPACKET 3
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{6E27CD90-8318-46C1-9520-F09DB7A90D0F}] DATAGRAM 3
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{CDBFE3B3-94AA-4000-9692-BFD57AB3A85C}] SEQPACKET 0
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No

Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{CDBFE3B3-94AA-4000-9692-BFD57AB3A85C}] DATAGRAM 0
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{6C298BF8-7CC7-4383-BC53-645051494892}] SEQPACKET 1
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{6C298BF8-7CC7-4383-BC53-645051494892}] DATAGRAM 1
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No

Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{0C65A653-F09B-4768-A08A-9D7CA9091FD0}] SEQPACKET 2
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{0C65A653-F09B-4768-A08A-9D7CA9091FD0}] DATAGRAM 2
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

[WinSock]

Item Value

[Ports]

[Serial]

Item Value

[Parallel]

Item Value

[Storage]

[Drives]

Item Value

[Disks]

Item Value

[SCSI]

Item Value

[IDE]

Item Value

[Printing]

Name Driver Port Name Server Name

[Problem Devices]

Device PNP Device ID Error Code

cLAN Host Adapter

PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&146CA173&0&18
This device is disabled.

cLAN Host Adapter #5

PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&300BC0BE&0&08
This device is disabled.

cLAN Host Adapter #2

PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&23FF515D&0&08
This device is disabled.

[USB]

Device PNP Device ID

Intel(r) 82371AB/EB PCI to USB Universal Host Controller

PCI\VEN_8086&DEV_7112&SUBSYS_00000000&REV_01\3&474B838&0&0A
USB Root Hub USB\ROOT_HUB\4&A0E0C64&0

[Software Environment]

[System Drivers]

Name	Description	File	Type	Started	Start Mode	State	Status
abiosdsk	Abiosdsk		Not Available	Kernel Driver	No		
acpi	Microsoft ACPI Driver	c:\windows\system32\drivers\acpi.sys	Kernel Driver	Yes	Boot	Running OK	Normal No Yes

acpiec	ACPIEC	c:\windows\system32\drivers\acpiec.sys	Kernel Driver	No	Disabled	Stopped OK	Normal No No
adpu160m	adpu160m	c:\windows\system32\drivers\adpu160m.sys	Kernel Driver	Yes	Boot	Running OK	Normal No Yes
afc9xxx	afc9xxx	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
afcnt	afcnt	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
afd	AFD Networking Support Environment	c:\windows\system32\drivers\afd.sys	Kernel Driver	Yes	Auto	Running OK	Normal No Yes
aha154x	Aha154x	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
aic78u2	aic78u2	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
aic78xx	aic78xx	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
aliide	AliIde	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
amsint	amsint	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
asc	asc	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
asc3550	asc3550	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
asynmac	RAS Asynchronous Media Driver	c:\windows\system32\drivers\asynmac.sys	Kernel Driver	No		Manual Stopped OK	Normal No No
atapi	atapi	c:\windows\system32\drivers\atapi.sys	Kernel Driver	No		Disabled	Stopped OK Normal No No
atdisk	Atdisk	Not Available	Kernel Driver	No	Disabled	Stopped OK	Ignore No No
atmarpc	ATM ARP Client Protocol	c:\windows\system32\drivers\atmarpc.sys	Kernel Driver	No		Manual Stopped OK	Normal No No
audstub	Audio Stub Driver	c:\windows\system32\drivers\audstub.sys	Kernel Driver	Yes		Manual Running OK	Normal No Yes
beep	Beep	c:\windows\system32\drivers\beep.sys	Kernel Driver	Yes		System Running OK	Normal No Yes
cbidf2k	cbidf2k	c:\windows\system32\drivers\cbidf2k.sys	Kernel Driver	No		Disabled	Stopped OK Normal No No
cdaudio	Cdaudio	c:\windows\system32\drivers\cdaudio.sys	Kernel Driver	No		System Stopped OK	Ignore No No
cdfs	Cdfs	c:\windows\system32\drivers\cdfs.sys	File System Driver	Yes		Disabled	Running OK Normal No Yes
cdrom	CD-ROM Driver	c:\windows\system32\drivers\cdrom.sys	Kernel Driver	Yes		System Running OK	Normal No Yes
changer	Changer	Not Available	Kernel Driver	No		Ignore No No	System Stopped OK
clusdisk	Cluster Disk Driver	c:\windows\system32\drivers\clusdisk.sys	Kernel Driver	No		Disabled	Stopped OK Normal No No
cmdide	CmdIde	Not Available	Kernel Driver	No	Disabled	Stopped OK	Normal No No
cpqarray	Cpqarray	Not Available	Kernel Driver	No		Disabled	Stopped OK Normal No No

cpqarray2	cpqarray2	Not Available	Kernel Driver	No		
	Disabled	Stopped OK	Normal	No	No	
cpqciissm	cpqciissm	Not Available	Kernel Driver	No		
	Disabled	Stopped OK	Normal	No	No	
cpqfcalm	cpqfcalm	Not Available	Kernel Driver	No		
	Disabled	Stopped OK	Normal	No	No	
dac2w2k	dac2w2k	c:\windows\system32\drivers\dac2w2k.sys	Kernel Driver			
	Yes	Boot Running OK	Normal	No	Yes	
dac960nt	dac960nt	Not Available	Kernel Driver	No		
	Disabled	Stopped OK	Normal	No	No	
dasd	Unisys DASD Driver v1.3.15	c:\windows\system32\drivers\dasd.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
dfsdriver	DfsDriver	c:\windows\system32\drivers\dfs.sys	File System Driver	Yes	Boot Running OK	Normal No Yes
disk	Disk Driver	c:\windows\system32\drivers\disk.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
dmboot	dmboot	c:\windows\system32\drivers\dmboot.sys	Kernel Driver	No	Disabled	Stopped OK Normal No No
dmio	Logical Disk Manager Driver	c:\windows\system32\drivers\dmio.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
dmload	dmload	c:\windows\system32\drivers\dmload.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
dpti2o	dpti2o	Not Available	Kernel Driver	No	Stopped OK	Normal No No
e100b	Intel(R) PRO Adapter Driver	c:\windows\system32\drivers\e100b325.sys	Kernel Driver	Yes	Manual Running OK	Normal No Yes
escon	ESCON	c:\windows\system32\drivers\escon.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
esvuart	Console Manager Network Connection	c:\windows\system32\drivers\esvuart.sys	Kernel Driver	Yes	Auto Running OK	Normal No Yes
fastfat	Fastfat	c:\windows\system32\drivers\fastfat.sys	File System Driver	Yes	Disabled	Running OK Normal No Yes
fdc	Floppy Disk Controller Driver	c:\windows\system32\drivers\fdc.sys	Kernel Driver	Yes	Manual Running OK	Normal No Yes
fips	Fips	c:\windows\system32\drivers\fips.sys	Kernel Driver	Yes	System Running OK	Normal No Yes
flpydisk	Floppy Disk Driver	c:\windows\system32\drivers\flpydisk.sys	Kernel Driver	Yes	Manual Running OK	Normal No Yes
ftdisk	Volume Manager Driver	c:\windows\system32\drivers\ftdisk.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
gnindis	cLAN NDIS Driver	c:\windows\system32\drivers\gnindis.sys	Kernel Driver	Yes	Auto Running OK	Normal No Yes
gnivia	cLAN VIA Driver	c:\windows\system32\drivers\gnivia.sys	Kernel Driver	Yes	Auto Running OK	Normal No Yes
gpc	Generic Packet Classifier	c:\windows\system32\drivers\msgpc.sys	Kernel Driver	Yes	Running OK	Normal No Yes
hpn	hpn	Not Available	Kernel Driver	No	Stopped OK	Normal No No

hpt3xx	hpt3xx	Not Available	Kernel Driver	No	Disabled	
http	HTTP Protocol Driver	c:\windows\system32\drivers\http.sys	Kernel Driver	Yes	Manual Running OK	Normal No Yes
i2omgmt	i2omgmt	Not Available	Kernel Driver	No	Normal No No	System Stopped OK
i2omp	i2omp	Not Available	Kernel Driver	No	Stopped OK	Normal No No
i2oubbs	Unisys Fibre Channel Driver v1.3.15	c:\windows\system32\drivers\i2oubbs.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver	c:\windows\system32\drivers\i8042prt.sys	Kernel Driver	Yes	System Running OK	Normal No Yes
imapi	Imapi	c:\windows\system32\drivers\imapi.sys	Kernel Driver	No	System Stopped OK	Ignore No No
ini910u	ini910u	Not Available	Kernel Driver	No	Stopped OK	Normal No No
intelide	IntelIde	c:\windows\system32\drivers\intelide.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
ipfilterdriver	IP Traffic Filter Driver	c:\windows\system32\drivers\ipfltdrv.sys	Kernel Driver	No	Manual Stopped OK	Normal No No
ipinip	IP in IP Tunnel Driver	c:\windows\system32\drivers\ipinip.sys	Kernel Driver	No	Manual Stopped OK	Normal No No
ipnat	IP Network Address Translator	c:\windows\system32\drivers\ipnat.sys	Kernel Driver	No	Manual Stopped OK	Normal No No
ipsec	IPSEC driver	c:\windows\system32\drivers\ipsec.sys	Kernel Driver	Yes	System Running OK	Normal No Yes
ipsraidn	ipsraidn	Not Available	Kernel Driver	No	Disabled	Stopped OK Normal No No
isapnp	PnP ISA/EISA Bus Driver	c:\windows\system32\drivers\isapnp.sys	Kernel Driver	Yes	Boot Running OK	Critical No Yes
kbdclass	Keyboard Class Driver	c:\windows\system32\drivers\kbdclass.sys	Kernel Driver	Yes	System Running OK	Normal No Yes
ksecdd	KSecDD	c:\windows\system32\drivers\ksecdd.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes
laguna	laguna	c:\windows\system32\drivers\cl546xm.sys	Kernel Driver	No	Manual Stopped OK	Ignore No No
lbtrfdc	lbtrfdc	Not Available	Kernel Driver	No	Ignore No No	System Stopped OK
lp6nds35	lp6nds35	Not Available	Kernel Driver	No	Disabled	Stopped OK Normal No No
mnmd	mnmd	c:\windows\system32\drivers\mnmd.sys	Kernel Driver	Yes	System Running OK	Ignore No Yes
modem	Modem	c:\windows\system32\drivers\modem.sys	Kernel Driver	No	Manual Stopped OK	Ignore No No
mouclass	Mouse Class Driver	c:\windows\system32\drivers\mouclass.sys	Kernel Driver	Yes	System Running OK	Normal No Yes
mountmgr	MountMgr	c:\windows\system32\drivers\mountmgr.sys	Kernel Driver	Yes	Boot Running OK	Normal No Yes

mraid35x	mraid35x	Not Available	Kernel Driver	No	
	Disabled	Stopped OK	Normal No	No	
mrxdav	WebDav Client Redirector				
	c:\windows\system32\drivers\mrxdav.sys		File System Driver		
	No	Manual Stopped OK	Normal No	No	
mrxsmb	MRXSMB	c:\windows\system32\drivers\mrxsmb.sys	File System		
Driver	Yes	System Running OK	Normal No	Yes	
msfs	MsfS	c:\windows\system32\drivers\msfs.sys	File System Driver		
	Yes	System Running OK	Normal No	Yes	
mup	Mup	c:\windows\system32\drivers\mup.sys	File System Driver		
	Yes	Boot Running OK	Normal No	Yes	
ndis	NDIS System Driver	c:\windows\system32\drivers\ndis.sys	Kernel		
Driver	Yes	Boot Running OK	Normal No	Yes	
ndistapi	Remote Access NDIS TAPI Driver				
	c:\windows\system32\drivers\ndistapi.sys		Kernel Driver	Yes	
	Manual Running OK	Normal No	Yes		
ndisuio	NDIS Usermode I/O Protocol				
	c:\windows\system32\drivers\ndisuio.sys		Kernel Driver	No	
	Manual Stopped OK	Normal No	No		
ndiswan	Remote Access NDIS WAN Driver				
	c:\windows\system32\drivers\ndiswan.sys		Kernel Driver	Yes	
	Manual Running OK	Normal No	Yes		
ndproxy	NDIS Proxy	c:\windows\system32\drivers\ndproxy.sys	Kernel		
Driver	Yes	Manual Running OK	Normal No	Yes	
netbios	NetBIOS Interface	c:\windows\system32\drivers\netbios.sys			
	File System Driver	Yes	System Running OK	Normal No	
	Yes				
netbt	NetBios over Tcpip	c:\windows\system32\drivers\netbt.sys	Kernel		
Driver	Yes	System Running OK	Normal No	Yes	
nfrd960	nfrd960	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
npfs	Npfs	c:\windows\system32\drivers\npfs.sys	File System Driver		
	Yes	System Running OK	Normal No	Yes	
ntfs	Ntfs	c:\windows\system32\drivers\ntfs.sys	File System Driver		
	Yes	Disabled Running OK	Normal No	Yes	
null	Null	c:\windows\system32\drivers\null.sys	Kernel Driver	No	
	System Stopped OK	Normal No	No		
nwlkflt	IPX Traffic Filter Driver				
	c:\windows\system32\drivers\nwlkflt.sys		Kernel Driver	No	
	Manual Stopped OK	Normal No	No		
nwlk fwd	IPX Traffic Forwarder Driver				
	c:\windows\system32\drivers\nwlk fwd.sys		Kernel Driver	No	
	Manual Stopped OK	Normal No	No		
parport	Parallel port driver	c:\windows\system32\drivers\parport.sys			
	Kernel Driver	Yes	Manual Running OK	Normal No	Yes
partmgr	PartMgr	c:\windows\system32\drivers\partmgr.sys	Kernel Driver		
	Yes	Boot Running OK	Normal No	Yes	
parvdm	ParVdm	c:\windows\system32\drivers\parvdm.sys	Kernel Driver		
	Yes	Auto Running OK	Ignore No	Yes	
pci	PCI Bus Driver	c:\windows\system32\drivers\pci.sys	Kernel Driver		
	Yes	Boot Running OK	Critical	No	Yes
pcidump	PCIDump	Not Available	Kernel Driver	No	System Stopped OK
	Ignore No	No			
pciide	PCIIde	Not Available	Kernel Driver	No	Disabled
	Stopped OK	Normal No	No		
pcmcia	Pcmcia	c:\windows\system32\drivers\pcmcia.sys	Kernel Driver		
	No	Disabled	Stopped OK	Normal No	No

pdcomp	PDCOMP	Not Available	Kernel Driver	No	Manual	Stopped OK
	Ignore No	No				
pdframe	PDFRAME	Not Available	Kernel Driver	No	Manual	Stopped OK
	Ignore No	No				
pdreli	PDRELI	Not Available	Kernel Driver	No	Manual	Stopped OK
	Ignore No	No				
pdrframe	PDRFRAME	Not Available	Kernel Driver	No	Manual	
	Stopped OK	Ignore No	No			
perc2	perc2	Not Available	Kernel Driver	No	Disabled	
	Stopped OK	Normal No	No			
perc2hib	perc2hib	Not Available	Kernel Driver	No		
	Disabled	Stopped OK	Normal No	No		
pptpminiport	WAN Miniport (PPTP)					
	c:\windows\system32\drivers\rasppptp.sys		Kernel Driver	Yes		
	Manual Running OK	Normal No	Yes			
processor	Processor Driver					
	c:\windows\system32\drivers\processr.sys		Kernel Driver	Yes		
	System Running OK	Normal No	Yes			
ptilink	Direct Parallel Link Driver					
	c:\windows\system32\drivers\ptilink.sys		Kernel Driver	Yes		
	Manual Running OK	Normal No	Yes			
ql1080	ql1080	Not Available	Kernel Driver	No	Disabled	
	Stopped OK	Normal No	No			
ql10wnt	Ql10wnt	Not Available	Kernel Driver	No	Disabled	
	Stopped OK	Normal No	No			
ql12160	ql12160	Not Available	Kernel Driver	No	Disabled	
	Stopped OK	Normal No	No			
ql1240	ql1240	Not Available	Kernel Driver	No	Disabled	
	Stopped OK	Normal No	No			
ql1280	ql1280	Not Available	Kernel Driver	No	Disabled	
	Stopped OK	Normal No	No			
ql2100	ql2100	Not Available	Kernel Driver	No	Disabled	
	Stopped OK	Normal No	No			
ql2200	ql2200	c:\windows\system32\drivers\ql2200.sys	Kernel Driver			
	Yes	Boot Running OK	Normal No	Yes		
rasacd	Remote Access Auto Connection Driver					
	c:\windows\system32\drivers\rasacd.sys		Kernel Driver	Yes		
	System Running OK	Normal No	Yes			
rasl2tp	WAN Miniport (L2TP)	c:\windows\system32\drivers\rasl2tp.sys				
	Kernel Driver	Yes	Manual Running OK	Normal No	Yes	
raspppoe	Remote Access PPPOE Driver					
	c:\windows\system32\drivers\raspppoe.sys		Kernel Driver	Yes		
	Manual Running OK	Normal No	Yes			
raspti	Direct Parallel	c:\windows\system32\drivers\raspti.sys				
	Kernel Driver	Yes	Manual Running OK	Normal No	Yes	
rdbss	Rdbss	c:\windows\system32\drivers\rdbss.sys	File System Driver			
	Yes	System Running OK	Normal No	Yes		
rdpcdd	RDPCDD	c:\windows\system32\drivers\rdpcdd.sys	Kernel Driver			
	Yes	System Running OK	Ignore No	Yes		
rdpdr	Terminal Server Device Redirector Driver					
	c:\windows\system32\drivers\rdpdr.sys		Kernel Driver	Yes	Manual	
	Running OK	Normal No	Yes			
rdpwd	RDPWD	c:\windows\system32\drivers\rdpwd.sys	Kernel Driver	Yes		
	Manual Running OK	Ignore No	Yes			
redbook	Digital CD Audio Playback Filter Driver					
	c:\windows\system32\drivers\redbook.sys		Kernel Driver	No		
	System Stopped OK	Normal No	No			


```

serenum Serenum Filter Driver c:\windows\system32\drivers\serenum.sys
Kernel Driver Yes Manual Running OK Normal No Yes

serial Serial port driver c:\windows\system32\drivers\serial.sys
Kernel Driver Yes System Running OK Ignore No Yes

sfloppy Sfloppy c:\windows\system32\drivers\sfloppy.sys Kernel Driver
No System Stopped OK Ignore No No

simbad Simbad Not Available Kernel Driver No Disabled
Stopped OK Normal No No

sparrow Sparrow Not Available Kernel Driver No Disabled
Stopped OK Normal No No

spud Special Purpose Utility Driver
c:\windows\system32\drivers\spud.sys Kernel Driver Yes Manual
Running OK Normal No Yes

srv Srv c:\windows\system32\drivers\srv.sys File System Driver
Yes Manual Running OK Normal No Yes

swenum Software Bus Driver c:\windows\system32\drivers\swenum.sys
Kernel Driver Yes Manual Running OK Normal No Yes

symc810 symc810 Not Available Kernel Driver No Disabled
Stopped OK Normal No No

symc8xx symc8xx Not Available Kernel Driver No Disabled
Stopped OK Normal No No

symmpi symmpi Not Available Kernel Driver No Disabled
Stopped OK Normal No No

sym_hi sym_hi Not Available Kernel Driver No Disabled
Stopped OK Normal No No

sym_u3 sym_u3 Not Available Kernel Driver No Disabled
Stopped OK Normal No No

tbs Tbs c:\windows\system32\drivers\tbs.sys Kernel Driver Yes
System Running OK Ignore No Yes

tcpip TCP/IP Protocol Driver c:\windows\system32\drivers\tcpip.sys Kernel
Driver Yes System Running OK Normal No Yes

tdasync TDASYNC c:\windows\system32\drivers\tdasync.sys Kernel Driver
No Manual Stopped OK Ignore No No

tdipx TDIPX c:\windows\system32\drivers\tdipx.sys Kernel Driver No
Manual Stopped OK Ignore No No

tdpipe TDPipe c:\windows\system32\drivers\tdpipe.sys Kernel Driver
No Manual Stopped OK Ignore No No

tdspx TDS PX c:\windows\system32\drivers\tdspx.sys Kernel Driver No
Manual Stopped OK Ignore No No

tdtcp TD TCP c:\windows\system32\drivers\tdtcp.sys Kernel Driver Yes
Manual Running OK Ignore No Yes

termdd Terminal Device Driver c:\windows\system32\drivers\termdd.sys
Kernel Driver Yes System Running OK Normal No Yes

toside TosIde Not Available Kernel Driver No Disabled
Stopped OK Normal No No

udfs Udfs c:\windows\system32\drivers\udfs.sys File System Driver
No Disabled Stopped OK Normal No No

unescons Console Manager Video Driver
c:\windows\system32\drivers\unescons.sys Kernel Driver Yes
Manual Running OK Ignore No Yes

update Microcode Update Driver
c:\windows\system32\drivers\update.sys Kernel Driver Yes
Manual Running OK Normal No Yes

```

```

usbhub USB2 Enabled Hub c:\windows\system32\drivers\usbhub.sys
Kernel Driver Yes Manual Running OK Normal No Yes

usbuhci Microsoft USB Universal Host Controller Miniport Driver
c:\windows\system32\drivers\usbuhci.sys Kernel Driver Yes
Manual Running OK Normal No Yes

vgasave VgaSave c:\windows\system32\drivers\vga.sys Kernel Driver Yes
System Running OK Ignore No Yes

viaide ViaIde Not Available Kernel Driver No Disabled
Stopped OK Normal No No

volsnap VolSnap c:\windows\system32\drivers\volsnap.sys Kernel Driver
Yes Boot Running OK Normal No Yes

vttsmon vttsmon c:\windows\system32\drivers\vttsmon.sys Kernel Driver
No Auto Stopped OK Normal No No

wanarp Remote Access IP ARP Driver
c:\windows\system32\drivers\wanarp.sys Kernel Driver Yes
Manual Running OK Normal No Yes

wdica WDICA Not Available Kernel Driver No Manual Stopped OK
Ignore No No

```

[Signed Drivers]

Device Name	Signed Manufacturer	Device Class	Driver Name	Version	Date
		INF Name			Device ID

[Environment Variables]

Variable	Value	User Name
ClusterLog	C:\WINDOWS\Cluster\cluster.log	<SYSTEM>
ComSpec	%SystemRoot%\system32\cmd.exe	<SYSTEM>
NUMBER_OF_PROCESSORS	32	<SYSTEM>
OS	Windows_NT	<SYSTEM>
Path	C:\Program Files\Intel\VTune6.0\CGGlbCache;C:\Program Files\Intel\VTune6.0\Analyzer\Bin\;C:\Program Files\Intel\VTune6.0\Shared\Bin\;%SystemRoot%\system32;%SystemRoot%\SystemRoot%\System32\Wbem;C:\Program Files\Microsoft SQL Server\80\Tools\BINN;C:\sql80\MSSQL\Binn	<SYSTEM>
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH	<SYSTEM>
PROCESSOR_ARCHITECTURE	x86	<SYSTEM>
PROCESSOR_IDENTIFIER	x86 Family 6 Model 10 Stepping 4, GenuineIntel	<SYSTEM>
PROCESSOR_LEVEL	6	<SYSTEM>
PROCESSOR_REVISION	0a04	<SYSTEM>
TEMP	%SystemRoot%\TEMP	<SYSTEM>
TMP	%SystemRoot%\TEMP	<SYSTEM>
windir	%SystemRoot%	<SYSTEM>
BISTRO_GLOBAL_CACHE_DIR	C:\Program Files\Intel\VTune6.0\CGGlbCache	<SYSTEM>
TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\LOCAL SERVICE
TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\LOCAL SERVICE
TEMP	%USERPROFILE%\Local Settings\Temp	MV-LE030P1\Administrator
TMP	%USERPROFILE%\Local Settings\Temp	MV-LE030P1\Administrator

[Print Jobs]

Document	Size	Owner	Notify	Status	Time Submitted	Start Time
Parameters	Until Time	Elapsed Time	Pages Printed	Job ID	Priority	
Data Type	Driver	Print Processor	Host	Print Queue		

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
------------	-------------	------	--------	-----------

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max
Working Set	Start Time	Version Size	File Date		
system idle process	Not Available	0	0	Not Available	Not Available
Available	Not Available	Not Available	Not Available	Not Available	Not Available
system	Not Available	4	8	0	1413120
Available	Not Available	Not Available	Not Available	Not Available	Not Available
smss.exe	c:\windows\system32\smss.exe	496	11	204800	
KB (44,544 bytes)	1413120 7/18/2001 12:35 PM	5.1.2496.0	(main.010615-1711)	43.50	
csrss.exe	c:\windows\system32\csrss.exe	544	13	Not Available	Not Available
7/18/2001 12:36 PM	Not Available	Not Available	Not Available	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	568	13	204800	
KB (427,008 bytes)	1413120 7/18/2001 12:36 PM	5.1.2496.0	(main.010615-1711)	417.00	
services.exe	c:\windows\system32\services.exe	612	9	204800	
KB (101,888 bytes)	1413120 7/18/2001 12:36 PM	5.1.2496.0	(main.010615-1711)	99.50	
lsass.exe	c:\windows\system32\lsass.exe	624	9	204800	
KB (11,776 bytes)	1413120 7/18/2001 12:36 PM	5.1.2496.0	(main.010615-1711)	11.50	
svchost.exe	c:\windows\system32\svchost.exe	832	8	204800	
KB (12,288 bytes)	1413120 7/18/2001 12:36 PM	5.1.2496.0	(main.010615-1711)	12.00	
gnconmgr.exe	c:\windows\system32\gnconmgr.exe	900	8	204800	
KB (131,119 bytes)	1413120 7/18/2001 12:36 PM	4.1.1	128.05	KB (131,119 bytes)	
svchost.exe	Not Available	944	8	Not Available	Not Available
7/18/2001 12:37 PM	Not Available	Not Available	Not Available	Not Available	Not Available
svchost.exe	c:\windows\system32\svchost.exe	964	8	204800	
KB (12,288 bytes)	1413120 7/18/2001 12:37 PM	5.1.2496.0	(main.010615-1711)	12.00	
esconmgr.exe	c:\windows\system32\esconmgr.exe	1036	24	204800	
KB (266,640 bytes)	1413120 7/18/2001 12:37 PM	6.0,2,9	260.39	KB (266,640 bytes)	
inetinfo.exe	c:\windows\system32\inetrv\inetinfo.exe	1216	8	204800	
KB (11,264 bytes)	1413120 7/18/2001 12:37 PM	6.0.2496.0	(main.010615-1711)	11.00	
proconsvr.exe	c:\windows\system32\proconsvr.exe	1248	8	204800	
bytes) 6/20/2001 12:46 PM	1413120 7/18/2001 12:37 PM	1, 0, 1, 30	80.00	KB (81,920	

proconmd8.exe	c:\windows\system32\proconmd8.exe	1352	8	204800	
bytes) 6/20/2001 12:46 PM	1413120 7/18/2001 12:37 PM	1, 0, 1, 30	31.00	KB (31,744	
svchost.exe	Not Available	1376	8	Not Available	Not Available
7/18/2001 12:37 PM	Not Available	Not Available	Not Available	Not Available	Not Available
svchost.exe	c:\windows\system32\svchost.exe	1624	8	204800	
KB (12,288 bytes)	1413120 7/18/2001 12:37 PM	5.1.2496.0	(main.010615-1711)	12.00	
svchost.exe	c:\windows\system32\svchost.exe	2040	8	204800	
KB (12,288 bytes)	1413120 7/18/2001 12:37 PM	5.1.2496.0	(main.010615-1711)	12.00	
csrss.exe	Not Available	344	13	Not Available	Not Available
7/18/2001 12:37 PM	Not Available	Not Available	Not Available	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	388	13	204800	
KB (427,008 bytes)	1413120 7/18/2001 12:37 PM	5.1.2496.0	(main.010615-1711)	417.00	
csrss.exe	Not Available	424	13	Not Available	Not Available
7/18/2001 12:37 PM	Not Available	Not Available	Not Available	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	448	13	204800	
KB (427,008 bytes)	1413120 7/18/2001 12:37 PM	5.1.2496.0	(main.010615-1711)	417.00	
explorer.exe	c:\windows\explorer.exe	3536	8	204800	
KB (1,000,448 bytes)	1413120 7/19/2001 8:48 AM	6.00.2496.0000	(main.010615-1711)	977.00	
isqlw.exe	c:\program files\microsoft sql server\80\tools\bin\isqlw.exe	3516	8	204800	1413120
bytes) 6/20/2001 3:25 PM	7/19/2001 6:44 PM	2000.080.0194.00	344.06	KB (352,319	
taskmgr.exe	c:\windows\system32\taskmgr.exe	2508	13	204800	
KB (128,512 bytes)	1413120 7/19/2001 7:33 PM	5.1.2496.0	(main.010615-1711)	125.50	
cmd.exe	c:\windows\system32\cmd.exe	2052	8	204800	1413120
(377,856 bytes)	7/19/2001 7:57 PM	5.1.2496.0	(main.010615-1711)	369.00	KB
msinfo32.exe	c:\program files\common files\microsoft shared\msinfo\msinfo32.exe	2992	8	204800	1413120 7/19/2001 7:58 PM
5.1.2496.0 (main.010615-1711)	39.00	KB (39,936 bytes)			
wmiprvse.exe	Not Available	1572	8	Not Available	Not Available
7/19/2001 7:58 PM	Not Available	Not Available	Not Available	Not Available	Not Available
msinfo32.exe	c:\program files\common files\microsoft shared\msinfo\msinfo32.exe	2264	8	204800	1413120 7/19/2001 7:59 PM
5.1.2496.0 (main.010615-1711)	39.00	KB (39,936 bytes)			

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer	Path
smss	5.1.2496.0	(main.010615-1711)	43.50	KB (44,544 bytes)	
	6/15/2001 8:23 PM			Microsoft Corporation	
				c:\windows\system32\smss.exe	
ntdll	5.1.2496.0	(main.010615-1711)	636.00	KB (651,264 bytes)	
	6/15/2001 8:18 PM			Microsoft Corporation	
				c:\windows\system32\ntdll.dll	

winlogon 5.1.2496.0 (main.010615-1711) 417.00 KB (427,008 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\winlogon.exe
kernel32 5.1.2496.0 (main.010615-1711) 875.50 KB (896,512 bytes)
6/15/2001 8:22 PM Microsoft Corporation
c:\windows\system32\kernel32.dll
advapi32 5.1.2496.0 (main.010615-1711) 535.00 KB (547,840 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\advapi32.dll
rpcrt4 5.1.2496.0 (main.010615-1711) 453.00 KB (463,872 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\rpcrt4.dll
authz 5.1.2496.0 (main.010615-1711) 49.00 KB (50,176 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\authz.dll
msvcrt 7.0.2496.0 (main.010615-1711) 315.00 KB (322,560 bytes)
6/15/2001 8:22 PM Microsoft Corporation
c:\windows\system32\msvcrt.dll
crypt32 5.1.2496.0 (main.010615-1711) 542.00 KB (555,008 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\crypt32.dll
user32 5.1.2496.0 (main.010615-1711) 545.00 KB (558,080 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\user32.dll
gdi32 5.1.2496.0 (main.010615-1711) 242.50 KB (248,320 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\gdi32.dll
msasn1 5.1.2496.0 (main.010615-1711) 50.00 KB (51,200 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\msasn1.dll
nddeapi 5.1.2496.0 (main.010615-1711) 15.00 KB (15,360 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\nddeapi.dll
profmap 5.1.2496.0 (main.010615-1711) 28.00 KB (28,672 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\profmap.dll
netapi32 5.1.2496.0 (main.010615-1711) 302.50 KB (309,760 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\netapi32.dll
userenv 5.1.2496.0 (main.010615-1711) 647.00 KB (662,528 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\userenv.dll
psapi 5.1.2496.0 (main.010615-1711) 17.00 KB (17,408 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\psapi.dll
regapi 5.1.2496.0 (main.010615-1711) 43.00 KB (44,032 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\regapi.dll
secur32 5.1.2496.0 (main.010615-1711) 50.50 KB (51,712 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\secur32.dll
setupapi 5.1.2496.0 (main.010615-1711) 916.00 KB (937,984 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\setupapi.dll
sfc_os 5.1.2496.0 (main.010615-1711) 129.00 KB (132,096 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\sfc_os.dll

wintrust 5.1.2496.0 (main.010615-1711) 163.50 KB (167,424 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\wintrust.dll
ole32 5.1.2496.0 (main.010615-1711) 1.09 MB (1,139,200 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\ole32.dll
imagehlp 5.1.2496.0 (main.010615-1711) 123.50 KB (126,464 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\imagehlp.dll
version 5.1.2496.0 (main.010615-1711) 16.00 KB (16,384 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\version.dll
winsta 5.1.2496.0 (main.010615-1711) 46.50 KB (47,616 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\winsta.dll
ws2_32 5.1.2496.0 (main.010615-1711) 73.50 KB (75,264 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\ws2_32.dll
ws2help 5.1.2496.0 (main.010615-1711) 18.50 KB (18,944 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\ws2help.dll
msgina 5.1.2496.0 (main.010615-1711) 937.00 KB (959,488 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\msgina.dll
shell32 6.00.2496.0000 (main.010615-1711) 7.94 MB (8,329,216 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\shell32.dll
shlwapi 6.00.2496.0000 (main.010615-1711) 386.00 KB (395,264 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\shlwapi.dll
comctl32 5.82 (main.010615-1711) 542.50 KB (555,520 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\comctl32.dll
odbc32 3.520.7601.1 196.00 KB (200,704 bytes) 6/16/2001 8:44 AM
Microsoft Corporation c:\windows\system32\odbc32.dll
comdlg32 6.00.2496.0000 (main.010615-1711) 252.00 KB (258,048 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\comdlg32.dll
comctl32 6.0 (main.010615-1711) 896.00 KB (917,504 bytes)
6/20/2001 5:33 AM Microsoft Corporation
c:\windows\winsxs\x86_microsoft.windows.common-
controls_6595b64144ccf1df_6.0.0.0_x-ww_1382d70a\comctl32.dll
odbcint 3.520.7601.1 88.00 KB (90,112 bytes) 6/6/2001 6:39 PM
Microsoft Corporation c:\windows\system32\odbcint.dll
shsvcs 6.00.2496.0000 (main.010615-1711) 112.00 KB (114,688 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\shsvcs.dll
sfc 5.1.2496.0 (main.010615-1711) 4.00 KB (4,096 bytes) 6/16/2001 8:44
AM Microsoft Corporation c:\windows\system32\sfc.dll
winscard 5.1.2496.0 (main.010615-1711) 91.00 KB (93,184 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\winscard.dll
wtsapi32 5.1.2496.0 (main.010615-1711) 16.50 KB (16,896 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\wtsapi32.dll
sxs 5.1.2496.0 (main.010615-1711) 596.50 KB (610,816 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\sxs.dll

rsaenh	5.1.2481.0 (Lab03_N.010523-1934)	128.50 KB (131,584 bytes)	6/15/2001 6:56 PM	Microsoft Corporation	c:\windows\system32\rsaenh.dll
wldap32	5.1.2496.0 (main.010615-1711)	164.50 KB (168,448 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\wldap32.dll
cscdll	5.1.2496.0 (main.010615-1711)	88.00 KB (90,112 bytes)	6/16/2001 8:43 AM	Microsoft Corporation	c:\windows\system32\cscdll.dll
wlnotify	5.1.2496.0 (main.010615-1711)	86.50 KB (88,576 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\wlnotify.dll
winmm	5.1.2496.0 (main.010615-1711)	165.50 KB (169,472 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\winmm.dll
winspool	5.1.2496.0 (main.010615-1711)	127.00 KB (130,048 bytes)	6/16/2001 8:46 AM	Microsoft Corporation	c:\windows\system32\winspool.drv
mpr	5.1.2496.0 (main.010615-1711)	55.00 KB (56,320 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\mpr.dll
uxtheme	6.00.2496.0000 (main.010615-1711)	198.00 KB (202,752 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\uxtheme.dll
samlib	5.1.2496.0 (main.010615-1711)	45.50 KB (46,592 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\samlib.dll
cscui	5.1.2496.0 (main.010615-1711)	302.50 KB (309,760 bytes)	6/16/2001 8:43 AM	Microsoft Corporation	c:\windows\system32\cscui.dll
iphlpapi	5.1.2496.2 (main.010615-1711)	75.00 KB (76,800 bytes)	6/16/2001 8:43 AM	Microsoft Corporation	c:\windows\system32\iphlpapi.dll
netman	5.1.2496.0 (main.010615-1711)	144.50 KB (147,968 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\netman.dll
mprapi	5.1.2496.0 (main.010615-1711)	77.50 KB (79,360 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\mprapi.dll
activeds	5.1.2496.0 (main.010615-1711)	177.50 KB (181,760 bytes)	6/16/2001 8:42 AM	Microsoft Corporation	c:\windows\system32\activeds.dll
adslldpc	5.1.2496.0 (main.010615-1711)	135.50 KB (138,752 bytes)	6/16/2001 8:42 AM	Microsoft Corporation	c:\windows\system32\adslldpc.dll
atl	3.00.9238	73.05 KB (74,802 bytes)	6/15/2001 8:21 PM	Microsoft Corporation	c:\windows\system32\atl.dll
oleaut32	3.50.5011.0	556.00 KB (569,344 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\oleaut32.dll
rtutils	5.1.2496.0 (main.010615-1711)	38.50 KB (39,424 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\rtutils.dll
rasapi32	5.1.2496.0 (main.010615-1711)	209.50 KB (214,528 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\rasapi32.dll
rasman	5.1.2496.0 (main.010615-1711)	54.50 KB (55,808 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\rasman.dll
tapi32	5.1.2496.0 (main.010615-1711)	160.00 KB (163,840 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\tapi32.dll
wzcsvc	5.1.2496.0 (main.010615-1711)	175.00 KB (179,200 bytes)	6/16/2001 1:45 AM	Microsoft Corporation	c:\windows\system32\wzcsvc.dll
wmi	5.1.2496.0 (main.010615-1711)	5.50 KB (5,632 bytes)	6/15/2001 8:18 PM	Microsoft Corporation	c:\windows\system32\wmi.dll
dhcpcsvc	5.1.2496.0 (main.010615-1711)	96.00 KB (98,304 bytes)	6/16/2001 8:43 AM	Microsoft Corporation	c:\windows\system32\dhcpcsvc.dll
dnsapi	5.1.2496.0 (main.010615-1711)	136.00 KB (139,264 bytes)	6/16/2001 8:43 AM	Microsoft Corporation	c:\windows\system32\dnsapi.dll
comres	2001.12.4414.12	773.00 KB (791,552 bytes)	6/16/2001 8:39 AM	Microsoft Corporation	c:\windows\system32\comres.dll
clbcatq	2001.12.4414.12	457.00 KB (467,968 bytes)	6/20/2001 12:46 PM	Microsoft Corporation	c:\windows\system32\clbcatq.dll
ntmarta	5.1.2496.0 (main.010615-1711)	107.00 KB (109,568 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\ntmarta.dll
wbemprox	5.1.2496.0 (main.010615-1711)	27.50 KB (28,160 bytes)	6/20/2001 12:45 PM	Microsoft Corporation	c:\windows\system32\wbem\wbemprox.dll
wbemcomn	5.1.2496.0 (main.010615-1711)	204.00 KB (208,896 bytes)	6/20/2001 12:45 PM	Microsoft Corporation	c:\windows\system32\wbem\wbemcomn.dll
wbemsvc	5.1.2496.0 (main.010615-1711)	47.00 KB (48,128 bytes)	6/20/2001 12:45 PM	Microsoft Corporation	c:\windows\system32\wbem\wbemsvc.dll
fastprox	5.1.2496.0 (main.010615-1711)	571.00 KB (584,704 bytes)	6/20/2001 12:45 PM	Microsoft Corporation	c:\windows\system32\wbem\fastprox.dll
services	5.1.2496.0 (main.010615-1711)	99.50 KB (101,888 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\services.exe
scserv	5.1.2496.0 (main.010615-1711)	359.00 KB (367,616 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\scserv.dll
umpnpgmr	5.1.2496.0 (main.010615-1711)	102.50 KB (104,960 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\umpnpgmr.dll
ncobjapi	5.1.2496.0 (main.010615-1711)	41.50 KB (42,496 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\ncobjapi.dll
eventlog	5.1.2496.0 (main.010615-1711)	46.50 KB (47,616 bytes)	6/16/2001 8:43 AM	Microsoft Corporation	c:\windows\system32\eventlog.dll
lsass	5.1.2496.0 (main.010615-1711)	11.50 KB (11,776 bytes)	6/16/2001 8:45 AM	Microsoft Corporation	c:\windows\system32\lsass.exe
lsasrv	5.1.2496.0 (main.010615-1711)	651.50 KB (667,136 bytes)	6/16/2001 8:43 AM	Microsoft Corporation	c:\windows\system32\lsasrv.dll
samsrv	5.1.2496.0 (main.010615-1711)	402.00 KB (411,648 bytes)	6/16/2001 8:44 AM	Microsoft Corporation	c:\windows\system32\samsrv.dll

```

cryptdll 5.1.2496.0 (main.010615-1711) 28.50 KB (29,184 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\cryptdll.dll
ntdsapi 5.1.2496.0 (main.010615-1711) 63.50 KB (65,024 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\ntdsapi.dll
msprivs 5.1.2496.0 (main.010615-1711) 44.00 KB (45,056 bytes)
6/15/2001 8:18 PM Microsoft Corporation
c:\windows\system32\msprivs.dll
kerberos 5.1.2496.0 (main.010615-1711) 259.00 KB (265,216 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\kerberos.dll
msvl_0 5.1.2496.0 (main.010615-1711) 106.00 KB (108,544 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\msvl_0.dll
netlogon 5.1.2496.0 (main.010615-1711) 389.50 KB (398,848 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\netlogon.dll
w32time 5.1.2496.0 (main.010615-1711) 159.50 KB (163,328 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\w32time.dll
msvcp60 6.00.8972.0 392.05 KB (401,462 bytes) 6/16/2001 8:39 AM
Microsoft Corporation c:\windows\system32\msvcp60.dll
schannel 5.1.2496.0 (main.010615-1711) 131.00 KB (134,144 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\schannel.dll
wdigest 5.1.2496.0 (main.010615-1711) 45.50 KB (46,592 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\wdigest.dll
rassfm 5.1.2496.0 (main.010615-1711) 19.50 KB (19,968 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\rassfm.dll
sfmapi 5.1.2496.0 (main.010615-1711) 22.50 KB (23,040 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\sfmapi.dll
kdcsvc 5.1.2496.0 (main.010615-1711) 168.50 KB (172,544 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\kdcsvc.dll
ntdsatq 5.1.2496.0 (main.010615-1711) 31.50 KB (32,256 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\ntdsatq.dll
mswsock 5.1.2496.0 (main.010615-1711) 225.50 KB (230,912 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\mswsock.dll
scecli 5.1.2496.0 (main.010615-1711) 170.50 KB (174,592 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\scecli.dll
ipsecsvc 5.1.2496.0 (main.010615-1711) 149.00 KB (152,576 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\ipsecsvc.dll
oakley 5.1.2496.0 (main.010615-1711) 724.00 KB (741,376 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\oakley.dll
winipsec 5.1.2496.0 (main.010615-1711) 25.00 KB (25,600 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\winipsec.dll
wshtcpip 5.1.2496.0 (main.010615-1711) 17.00 KB (17,408 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\wshtcpip.dll

```

```

dssenh 5.1.2481.0 (Lab03_N.010523-1934) 121.00 KB (123,904 bytes)
6/15/2001 6:56 PM Microsoft Corporation
c:\windows\system32\dssenh.dll
pstorsvc 5.1.2496.0 (main.010615-1711) 25.00 KB (25,600 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\pstorsvc.dll
psbase 5.1.2496.0 (main.010615-1711) 80.50 KB (82,432 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\psbase.dll
svchost 5.1.2496.0 (main.010615-1711) 12.00 KB (12,288 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\svchost.exe
rpcss 5.1.2496.0 (main.010615-1711) 252.50 KB (258,560 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\rpcss.dll
winrnr 5.1.2496.0 (main.010615-1711) 14.50 KB (14,848 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\winrnr.dll
rasadhlp 5.1.2496.0 (main.010615-1711) 6.00 KB (6,144 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\rasadhlp.dll
gnconmgr 4.1.1 128.05 KB (131,119 bytes) 6/21/2001 4:21 PM
Giganet Incorporated c:\windows\system32\gnconmgr.exe
vipl 4.1.1 80.00 KB (81,920 bytes) 6/25/2001 1:05 PM
Giganet Incorporated c:\windows\system32\vipl.dll
wkssvc 5.1.2496.0 (main.010615-1711) 118.50 KB (121,344 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\wkssvc.dll
audiosrv 5.1.2496.0 (main.010615-1711) 37.00 KB (37,888 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\audiosrv.dll
dmserver 2496.0.503.0 21.00 KB (21,504 bytes) 6/16/2001 8:43
AM Microsoft Corp. c:\windows\system32\dmserver.dll
ersvc 5.1.2496.0 (main.010615-1711) 17.00 KB (17,408 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\ersvc.dll
es 2001.12.4414.12 219.50 KB (224,768 bytes) 6/16/2001 8:43
AM Microsoft Corporation c:\windows\system32\es.dll
srvsvc 5.1.2496.0 (main.010615-1711) 85.00 KB (87,040 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\srvsvc.dll
seclogon 5.1.2496.0 (main.010615-1711) 16.00 KB (16,384 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\seclogon.dll
sens 5.1.2496.0 (main.010615-1711) 35.00 KB (35,840 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\sens.dll
comsvcs 2001.12.4414.12 1.10 MB (1,158,144 bytes) 6/20/2001
12:46 PM Microsoft Corporation c:\windows\system32\comsvcs.dll
mtxclu 2001.12.4414.12 60.00 KB (61,440 bytes) 6/16/2001 8:44
AM Microsoft Corporation c:\windows\system32\mtxclu.dll
wsock32 5.1.2496.0 (main.010615-1711) 21.00 KB (21,504 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\wsock32.dll
colbact 2001.12.4414.12 66.50 KB (68,096 bytes) 6/20/2001
12:46 PM Microsoft Corporation c:\windows\system32\colbact.dll

```

```

clusapi 5.1.2496.0 (main.010615-1711) 52.00 KB (53,248 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\clusapi.dll
resutils 5.1.2496.0 (main.010615-1711) 53.50 KB (54,784 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\resutils.dll
mtxoci 2001.12.4414.12 82.00 KB (83,968 bytes) 6/20/2001
12:46 PM Microsoft Corporation c:\windows\system32\mtxoci.dll

wmisvc 5.1.2496.0 (main.010615-1711) 99.00 KB (101,376 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\wmisvc.dll
vssapi 5.1.2496.0 (main.010615-1711) 524.00 KB (536,576 bytes)
6/16/2001 8:41 AM Microsoft Corporation
c:\windows\system32\vssapi.dll
wbemcore 5.1.2496.0 (main.010615-1711) 465.00 KB (476,160 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\wbemcore.dll
esscli 5.1.2496.0 (main.010615-1711) 230.00 KB (235,520 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\esscli.dll
wmiutils 5.1.2496.0 (main.010615-1711) 93.50 KB (95,744 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\wmiutils.dll
repdrvfs 5.1.2496.0 (main.010615-1711) 132.50 KB (135,680 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\repdrvfs.dll
wmiprvsd 5.1.2496.0 (main.010615-1711) 396.00 KB (405,504 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\wmiprvsd.dll
wbemess 5.1.2496.0 (main.010615-1711) 253.00 KB (259,072 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\wbemess.dll
ncprov 5.1.2496.0 (main.010615-1711) 59.50 KB (60,928 bytes)
6/20/2001 12:45 PM Microsoft Corporation
c:\windows\system32\wbem\ncprov.dll
netshell 5.1.2496.0 (main.010615-1711) 1.54 MB (1,617,920 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\netshell.dll
credui 5.1.2496.0 (main.010615-1711) 158.00 KB (161,792 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\credui.dll
hnetcfg 5.1.2496.0 (main.010615-1711) 236.50 KB (242,176 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\hnetcfg.dll
wininet 6.0.2496.0000 (main.010615-1711) 580.00 KB (593,920 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\wininet.dll
netcfgx 5.1.2496.0 (main.010615-1711) 570.50 KB (584,192 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\netcfgx.dll
msi 2.0.2496.0 2.00 MB (2,102,272 bytes) 6/16/2001 8:44 AM
Microsoft Corporation c:\windows\system32\msi.dll
rasdlg 5.1.2496.0 (main.010615-1711) 617.00 KB (631,808 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\rasdlg.dll
ntlsapi 5.1.2496.0 (main.010615-1711) 6.00 KB (6,144 bytes) 6/16/2001 8:44
AM Microsoft Corporation c:\windows\system32\ntlsapi.dll

```

```

cryptsvc 5.1.2496.0 (main.010615-1711) 50.00 KB (51,200 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\cryptsvc.dll
certcli 5.1.2496.0 (main.010615-1711) 180.00 KB (184,320 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\certcli.dll
cryptui 5.131.2496.0 (main.010615-1711) 459.00 KB (470,016 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\cryptui.dll
esent 5.1.2496.0 (main.010615-1711) 1,000.50 KB (1,024,512 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\esent.dll
cabinet 5.1.2496.0 (main.010615-1711) 57.50 KB (58,880 bytes)
6/16/2001 8:42 AM Microsoft Corporation
c:\windows\system32\cabinet.dll
escommgr 6,0,2,9 260.39 KB (266,640 bytes) 5/3/2001 12:59 PM
Unisys c:\windows\system32\escommgr.exe
inetinfo 6.0.2496.0 (main.010615-1711) 11.00 KB (11,264 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\inetinfo.exe
rpcref 6.0.2496.0 (main.010615-1711) 4.00 KB (4,096 bytes) 6/20/2001
12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\rpcref.dll
iisrtl 6.0.2496.0 (main.010615-1711) 118.00 KB (120,832 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\iisrtl.dll
iisadmin 6.0.2496.0 (main.010615-1711) 21.00 KB (21,504 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\iisadmin.dll
coadmin 6.0.2496.0 (main.010615-1711) 41.50 KB (42,496 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\coadmin.dll
admwprox 6.0.2496.0 (main.010615-1711) 34.50 KB (35,328 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\admwprox.dll
metadata 6.0.2496.0 (main.010615-1711) 209.00 KB (214,016 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\metadata.dll
iiscfg 1.0.2916.0 1.50 MB (1,577,984 bytes) 6/20/2001 12:46 PM
Microsoft Corporation c:\windows\system32\inetinfo\iiscfg.dll

msxml3 8.20.8608.0 1.06 MB (1,111,552 bytes) 6/16/2001 8:44 AM
Microsoft Corporation c:\windows\system32\msxml3.dll
nsepm 6.0.2496.0 (main.010615-1711) 40.50 KB (41,472 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\nsepm.dll
iismap 6.0.2496.0 (main.010615-1711) 59.00 KB (60,416 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\iismap.dll
sslcfg 6.0.2496.0 (main.010615-1711) 17.50 KB (17,920 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\sslcfg.dll
iisutil 6.0.2496.0 (main.010615-1711) 113.00 KB (115,712 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\iisutil.dll
wamreg 6.0.2496.0 (main.010615-1711) 42.50 KB (43,520 bytes)
6/20/2001 12:46 PM Microsoft Corporation
c:\windows\system32\inetinfo\wamreg.dll

```

admexs 6.0.2496.0 (main.010615-1711) 27.00 KB (27,648 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\admexs.dll
 svcxext 6.0.2496.0 (main.010615-1711) 45.50 KB (46,592 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\svcxext.dll
 security 5.1.2496.0 (main.010615-1711) 5.00 KB (5,120 bytes)
 6/16/2001 8:39 AM Microsoft Corporation
 c:\windows\system32\security.dll
 smtpsvc 6.0.2496.0 (main.010615-1711) 421.50 KB (431,616 bytes)
 6/20/2001 12:50 PM Microsoft Corporation
 c:\windows\system32\inetsrv\smtpsvc.dll
 infocomm 6.0.2496.0 (main.010615-1711) 235.00 KB (240,640 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\infocomm.dll
 isatq 6.0.2496.0 (main.010615-1711) 63.50 KB (65,024 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\isatq.dll
 iisfecnv 6.0.2496.0 (main.010615-1711) 6.50 KB (6,656 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\iisfecnv.dll
 fcachdll 6.0.2496.0 (main.010615-1711) 42.50 KB (43,520 bytes)
 6/20/2001 12:50 PM Microsoft Corporation
 c:\windows\system32\fcachdll.dll
 rwnh 6.0.2496.0 (main.010615-1711) 9.00 KB (9,216 bytes) 6/20/2001
 12:50 PM Microsoft Corporation c:\windows\system32\rwnh.dll
 exstrace 5.6.2496.0 (main.010615-1711) 13.00 KB (13,312 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\exstrace.dll
 staxmem 6.0.2496.0 (main.010615-1711) 7.50 KB (7,680 bytes) 6/20/2001
 12:46 PM Microsoft Corporation c:\windows\system32\staxmem.dll
 lonsint 6.0.2496.0 (main.010615-1711) 11.00 KB (11,264 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\lonsint.dll
 iscomlog 6.0.2496.0 (main.010615-1711) 23.50 KB (24,064 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\iscomlog.dll
 seo 6.0.2496.0 (main.010615-1711) 201.00 KB (205,824 bytes)
 6/20/2001 12:50 PM Microsoft Corporation
 c:\windows\system32\inetsrv\seo.dll
 aqueue 6.0.2496.0 (main.010615-1711) 305.00 KB (312,320 bytes)
 6/20/2001 12:50 PM Microsoft Corporation
 c:\windows\system32\inetsrv\aqueue.dll
 ntfsdrv 6.0.2496.0 (main.010615-1711) 38.00 KB (38,912 bytes)
 6/20/2001 12:50 PM Microsoft Corporation
 c:\windows\system32\inetsrv\ntfsdrv.dll
 proconsv 1, 0, 1, 30 80.00 KB (81,920 bytes) 6/20/2001
 12:46 PM IBM Corporation c:\windows\system32\proconsv.exe
 proconmd 1, 0, 1, 30 31.00 KB (31,744 bytes) 6/20/2001
 12:46 PM IBM Corporation c:\windows\system32\proconmd.exe
 w3ssl 6.0.2496.0 (main.010615-1711) 11.50 KB (11,776 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\w3ssl.dll
 strmfilt 6.0.2496.0 (main.010615-1711) 31.50 KB (32,256 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\inetsrv\strmfilt.dll

httpapi 6.0.2496.0 (main.010615-1711) 9.00 KB (9,216 bytes) 6/20/2001
 12:46 PM Microsoft Corporation c:\windows\system32\httpapi.dll
 w3tp 6.0.2496.0 (main.010615-1711) 9.00 KB (9,216 bytes) 6/20/2001
 12:46 PM Microsoft Corporation c:\windows\system32\inetsrv\w3tp.dll
 termsrv 5.1.2496.0 (main.010615-1711) 192.50 KB (197,120 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\termsrv.dll
 icaapi 5.1.2496.0 (main.010615-1711) 8.50 KB (8,704 bytes) 6/20/2001
 12:46 PM Microsoft Corporation c:\windows\system32\icaapi.dll
 mstlsapi 5.1.2496.0 (main.010615-1711) 101.00 KB (103,424 bytes)
 6/16/2001 8:44 AM Microsoft Corporation
 c:\windows\system32\mstlsapi.dll
 rdpwsx 5.1.2496.0 (main.010615-1711) 72.13 KB (73,864 bytes)
 6/20/2001 12:46 PM Microsoft Corporation
 c:\windows\system32\rdpwsx.dll
 explorer 6.0.2496.0000 (main.010615-1711) 977.00 KB (1,000,448
 bytes) 6/16/2001 8:45 AM Microsoft Corporation
 c:\windows\explorer.exe
 browseui 6.0.2496.0000 (main.010615-1711) 1,003.50 KB
 (1,027,584 bytes) 6/16/2001 8:42 AM Microsoft Corporation
 c:\windows\system32\browseui.dll
 shdocvw 6.0.2496.0000 (main.010615-1711) 1.28 MB (1,344,512 bytes)
 6/16/2001 8:44 AM Microsoft Corporation
 c:\windows\system32\shdocvw.dll
 apphelp 5.1.2496.0 (main.010615-1711) 98.50 KB (100,864 bytes)
 6/16/2001 8:42 AM Microsoft Corporation
 c:\windows\system32\apphelp.dll
 themeui 6.0.2496.0000 (main.010615-1711) 376.50 KB (385,536 bytes)
 6/16/2001 8:45 AM Microsoft Corporation
 c:\windows\system32\themeui.dll
 msimg32 5.1.2496.0 (main.010615-1711) 4.50 KB (4,608 bytes) 6/16/2001 8:44
 AM Microsoft Corporation c:\windows\system32\msimg32.dll
 linkinfo 5.1.2496.0 (main.010615-1711) 15.50 KB (15,872 bytes)
 6/16/2001 8:43 AM Microsoft Corporation
 c:\windows\system32\linkinfo.dll
 ntshrui 5.1.2496.0 (main.010615-1711) 132.50 KB (135,680 bytes)
 6/16/2001 8:44 AM Microsoft Corporation
 c:\windows\system32\ntshrui.dll
 webcheck 6.0.2496.0000 (main.010615-1711) 253.00 KB (259,072
 bytes) 6/16/2001 8:45 AM Microsoft Corporation
 c:\windows\system32\webcheck.dll
 stobject 5.1.2496.0 (main.010615-1711) 114.50 KB (117,248 bytes)
 6/16/2001 8:45 AM Microsoft Corporation
 c:\windows\system32\stobject.dll
 batmeter 6.0.2496.0000 (main.010615-1711) 26.50 KB (27,136
 bytes) 6/16/2001 8:42 AM Microsoft Corporation
 c:\windows\system32\batmeter.dll
 powrprof 6.0.2496.0000 (main.010615-1711) 14.00 KB (14,336
 bytes) 6/16/2001 8:44 AM Microsoft Corporation
 c:\windows\system32\powrprof.dll
 browselc 6.0.2496.0000 (main.010615-1711) 61.50 KB (62,976
 bytes) 6/16/2001 8:39 AM Microsoft Corporation
 c:\windows\system32\browselc.dll
 shdoclc 6.0.2496.0000 (main.010615-1711) 536.00 KB (548,864 bytes)
 6/15/2001 8:18 PM Microsoft Corporation
 c:\windows\system32\shdoclc.dll

```

drprov 5.1.2496.0 (main.010615-1711) 11.50 KB (11,776 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\drprov.dll
ntlanman 5.1.2496.0 (main.010615-1711) 37.50 KB (38,400 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\ntlanman.dll
netui0 5.1.2496.0 (main.010615-1711) 73.00 KB (74,752 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\netui0.dll
netui1 5.1.2496.0 (main.010615-1711) 224.50 KB (229,888 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\netui1.dll
netrap 5.1.2496.0 (main.010615-1711) 10.50 KB (10,752 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\netrap.dll
davclnt 5.1.2496.0 (main.010615-1711) 22.00 KB (22,528 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\davclnt.dll
printui 5.1.2496.0 (main.010615-1711) 510.50 KB (522,752 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\printui.dll
cfgmgr32 5.1.2496.0 (main.010615-1711) 16.00 KB (16,384 bytes)
6/15/2001 8:18 PM Microsoft Corporation
c:\windows\system32\cfgmgr32.dll
urlmon 6.00.2496.0000 (main.010615-1711) 473.00 KB (484,352 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\urlmon.dll
isqlw 2000.080.0194.00 344.06 KB (352,319 bytes) 6/20/2001 3:25
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\isqlw.exe
sqlunirl 2000.080.0380.00 176.56 KB (180,800 bytes)
6/16/2001 8:41 AM Microsoft Corporation
c:\windows\system32\sqlunirl.dll
sqlresld 2000.080.0194.00 28.06 KB (28,738 bytes)
6/20/2001 3:26 PM Microsoft Corporation c:\program
files\microsoft sql server\80\tools\bin\sqlresld.dll
sqlgui 2000.080.0194.00 444.06 KB (454,720 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\sqlgui.dll
w95scm 2000.080.0194.00 48.06 KB (49,216 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\w95scm.dll
sqlsvc 2000.080.0194.00 92.06 KB (94,272 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\sqlsvc.dll
odbcbc 2000.081.7601.00 24.00 KB (24,576 bytes) 6/16/2001 8:44
AM Microsoft Corporation c:\windows\system32\odbcbc.dll
imm32 5.1.2496.0 (main.010615-1711) 95.50 KB (97,792 bytes)
6/16/2001 8:43 AM Microsoft Corporation
c:\windows\system32\imm32.dll
sqlqry 2000.080.0194.00 392.06 KB (401,472 bytes) 6/20/2001 3:25
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\sqlqry.dll
pfutil80 2000.080.0194.00 268.06 KB (274,498 bytes)
6/20/2001 3:26 PM Microsoft Corporation c:\program
files\microsoft sql server\80\tools\bin\pfutil80.dll
semsfc 2000.080.0194.00 224.06 KB (229,440 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\semsfc.dll

```

```

pfclnt80 2000.080.0194.00 404.06 KB (413,762 bytes)
6/20/2001 3:26 PM Microsoft Corporation c:\program
files\microsoft sql server\80\tools\bin\pfclnt80.dll
tsappcmp 5.1.2496.0 (main.010615-1711) 51.50 KB (52,736 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\tsappcmp.dll
sqlsvc 2000.080.0194.00 24.00 KB (24,576 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlsvc.rll
sqlgui 2000.080.0194.00 56.00 KB (57,344 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlgui.rll
semsfc 2000.080.0194.00 24.00 KB (24,576 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\resources\1033\semsfc.rll
pfclnt80 2000.080.0194.00 28.00 KB (28,672 bytes)
6/20/2001 3:26 PM Microsoft Corporation c:\program
files\microsoft sql server\80\tools\bin\resources\1033\pfclnt80.rll
pfutil80 2000.080.0194.00 144.00 KB (147,456 bytes)
6/20/2001 3:26 PM Microsoft Corporation c:\program
files\microsoft sql server\80\tools\bin\resources\1033\pfutil80.rll
sqlqry 2000.080.0194.00 180.00 KB (184,320 bytes) 6/20/2001 3:25
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlqry.rll
isqlw 2000.080.0194.00 240.00 KB (245,760 bytes) 6/20/2001 3:25
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\resources\1033\isqlw.rll
sqlsrv32 2000.081.7601.00 348.00 KB (356,352 bytes)
6/6/2001 6:39 PM Microsoft Corporation
c:\windows\system32\sqlsrv32.dll
sqlsrv32 2000.081.7601.00 88.00 KB (90,112 bytes)
6/6/2001 6:39 PM Microsoft Corporation
c:\windows\system32\sqlsrv32.rll
odbcpc 3.520.7601.1 92.00 KB (94,208 bytes) 6/16/2001 8:44
AM Microsoft Corporation c:\windows\system32\odbcpc.dll
sqllex 2000.080.0194.00 148.06 KB (151,616 bytes) 6/20/2001 3:26
PM Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\sqllex.dll
dbnetlib 2000.081.7601.00 64.00 KB (65,536 bytes) 6/16/2001 8:40
AM Microsoft Corporation c:\windows\system32\dbnetlib.dll
dbmslpcn 2000.080.0194.00 28.06 KB (28,734 bytes)
6/20/2001 3:24 PM Microsoft Corporation
c:\windows\system32\dbmslpcn.dll
taskmgr 5.1.2496.0 (main.010615-1711) 125.50 KB (128,512 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\taskmgr.exe
vdmdbg 5.1.2496.0 (main.010615-1711) 24.00 KB (24,576 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\vdmdbg.dll
utildll 5.1.2496.0 (main.010615-1711) 25.00 KB (25,600 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\utildll.dll
cmd 5.1.2496.0 (main.010615-1711) 369.00 KB (377,856 bytes)
6/16/2001 8:45 AM Microsoft Corporation
c:\windows\system32\cmd.exe

```



```

msinfo32      5.1.2496.0 (main.010615-1711) 39.00 KB (39,936 bytes)
6/20/2001 12:53 PM Microsoft Corporation c:\program
files\common files\microsoft shared\msinfo\msinfo32.exe
mfc42u 6.00.8665.0 972.05 KB (995,384 bytes) 6/16/2001 8:44 AM
Microsoft Corporation c:\windows\system32\mfc42u.dll
msinfo 7, 0, 0 339.50 KB (347,648 bytes) 6/20/2001 12:54 PM
c:\windows\pchealth\helpctr\binaries\msinfo.dll
riched32      5.1.2496.0 (main.010615-1711) 3.50 KB (3,584 bytes)
6/16/2001 8:44 AM Microsoft Corporation
c:\windows\system32\riched32.dll
riched20      5.30.23.1210 414.50 KB (424,448 bytes) 6/16/2001 8:44
AM Microsoft Corporation c:\windows\system32\riched20.dll

```

[Services]

Display Name	Name	State	Start Mode	Service Type	Path	Error
Control	Start Name	Tag ID				
Alerter	Alerter	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k localservice	Normal NT
AUTHORITY\LocalService 0						
Application Management	AppMgmt	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
ASP.NET State aspnet_state Stopped Manual Own Process						
c:\windows\microsoft.net\Framework\v1.0.2817\aspnet_state.exe						
Normal LocalSystem 0						
Windows Audio	AudioSrv	Running Auto	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Background Intelligent Transfer Service		Stopped Manual	Share Process	BITS	c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Computer Browser	Browser	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Indexing Service	cisvc	Stopped Manual	Share Process		c:\windows\system32\cisvc.exe	Normal LocalSystem 0
ClipBook	ClipSrv	Stopped Manual	Own Process		c:\windows\system32\clipsrv.exe	Normal LocalSystem 0
COM+ System Application	COMSysApp	Stopped Manual	Own Process		c:\windows\system32\dlhhost.exe /processid:{02d4b3f1-fd88-11d1-960d-00805fc79235}	Normal LocalSystem 0
.NET Framework Support Service	CORRTSvc	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k corsvcs	Normal LocalSystem 0
Cryptographic Services	CryptSvc	Running Manual	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Distributed File System	Dfs	Stopped Manual	Own Process		c:\windows\system32\dfssvc.exe	Normal LocalSystem 0
DHCP Client	Dhcp	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Logical Disk Manager	Administrative Service	Stopped Manual	Share Process		c:\windows\system32\dmadmin.exe /com	Normal LocalSystem 0

Logical Disk Manager	dmserver	Running Auto	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
DNS Client	Dnscache	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k networkservice	Normal NT
AUTHORITY\NetworkService 0						
Error Reporting Service	ERSvc	Running Auto	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Ignore LocalSystem 0
Console Manager	esConMgr	Running Auto	Own Process		c:\windows\system32\esconmgr.exe	Normal LocalSystem 0
Event Log	Eventlog	Running Auto	Share Process		c:\windows\system32\services.exe	Normal LocalSystem 0
COM+ Event System	EventSystem	Running Manual	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Fast User Switching Compatibility FastUserSwitchingCompatibility						
Stopped Manual Share Process c:\windows\system32\svchost.exe -k						
netsvcs	Normal	LocalSystem	0			
cLAN Connection Manager	GniConMgr	Running Auto	Own Process		c:\windows\system32\gnconmgr.exe	Normal LocalSystem 0
Help and Support	helpsvc	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Human Interface Device Access	HidServ	Stopped Disabled	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
IIS Admin	IISADMIN	Running Manual	Share Process		c:\windows\system32\inethttpd.exe	Normal LocalSystem 0
IMAPI CD-Burning COM Service	ImapiService	Stopped Manual	Own Process		c:\windows\system32\imapi.exe	Normal LocalSystem 0
Intersite Messaging	IsmServ	Stopped Disabled	Own Process		c:\windows\system32\ismserv.exe	Normal LocalSystem 0
Kerberos Key Distribution Center	kdc	Stopped Disabled	Share Process		c:\windows\system32\lsass.exe	Normal LocalSystem 0
Server	lanmanserver	Running Auto	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
Workstation	lanmanworkstation	Running Auto	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
License Logging	LicenseService	Stopped Manual	Own Process		c:\windows\system32\llssrv.exe	Normal NT
AUTHORITY\NetworkService 0						
TCP/IP NetBIOS Helper	LmHosts	Running Auto	Share Process		c:\windows\system32\svchost.exe -k localservice	Normal NT
AUTHORITY\LocalService 0						
Messenger	Messenger	Stopped Manual	Share Process		c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem 0
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped Manual	Own Process		c:\windows\system32\mnmsrvc.exe	Normal LocalSystem 0

Distributed Transaction Coordinator	MSDTC	Stopped Manual	Own Process
c:\windows\system32\msdtc.exe Normal NT AUTHORITY\NetworkService 0			
Windows Installer	MSIServer	Stopped Manual	Share Process
c:\windows\system32\msiexec.exe /v Normal LocalSystem 0			
MSSQLSERVER	MSSQLSERVER	Stopped Manual	Own Process
c:\sql80\mssql\binn\sqlservr.exe Normal LocalSystem 0			
MSSQLServerADHelper	MSSQLServerADHelper	Stopped Manual	Own Process
c:\program files\microsoft sql server\80\tools\binn\sqladhlp.exe Normal LocalSystem 0			
Navisphere Agent	Navisphere_Agent	Stopped Manual	Own Process
c:\program files\unisys\navisphere agent\naviagent.exe Normal LocalSystem 0			
Network DDE	NetDDE	Stopped Manual	Share Process
c:\windows\system32\netdde.exe Normal LocalSystem 0			
Network DDE DSDM	NetDDEdsdm	Stopped Manual	Share Process
c:\windows\system32\netdde.exe Normal LocalSystem 0			
Net Logon	Netlogon	Stopped Manual	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0			
Network Connections	Netman	Running Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			
Network Location Awareness (NLA)	Nla	Running Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			
File Replication	NtFrs	Stopped Manual	Own Process
c:\windows\system32\ntfrs.exe Ignore LocalSystem 0			
NT LM Security Support Provider	NtLmSsp	Stopped Manual	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0			
Removable Storage	NtmsSvc	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			
Plug and Play	PlugPlay	Running Auto	Share Process
c:\windows\system32\services.exe Normal LocalSystem 0			
IPSEC Services PolicyAgent		Running Auto	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0			
Process Control Service	ProcCon	Running Auto	Own Process
c:\windows\system32\proconsvcs.exe Normal LocalSystem 0			
Protected Storage	ProtectedStorage	Running Auto	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0			
Remote Access Auto Connection Manager	RasAuto	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			
Remote Access Connection Manager	RasMan	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			
Remote Desktop Help Session Manager	RDSessMgr	Stopped Manual	Own Process
c:\windows\system32\sessmgr.exe Normal LocalSystem 0			
Routing and Remote Access	RemoteAccess	Stopped Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			

Remote Registry	RemoteRegistry	Running Auto	Share Process
c:\windows\system32\svchost.exe -k regsvcs Normal NT AUTHORITY\LocalService 0			
Remote Procedure Call (RPC) Locator	RpcLocator	Stopped Manual	Own Process
c:\windows\system32\locator.exe Normal NT AUTHORITY\NetworkService 0			
Remote Procedure Call (RPC)	RpcSs	Running Auto	Share Process
c:\windows\system32\svchost -k rpcss Normal LocalSystem 0			
Resultant Set of Policy Provider	RSOPProv	Stopped Manual	Share Process
c:\windows\system32\rsopprov.exe Normal LocalSystem 0			
QoS RSVP	RSVP	Stopped Manual	Own Process
c:\windows\system32\rsvp.exe Normal LocalSystem 0			
Security Accounts Manager	SamSs	Running Auto	Share Process
c:\windows\system32\lsass.exe Normal LocalSystem 0			
Smart Card Helper	SCardDrv	Stopped Manual	Share Process
c:\windows\system32\scardsvr.exe Ignore NT AUTHORITY\LocalService 0			
Smart Card	SCardSvr	Stopped Manual	Share Process
c:\windows\system32\scardsvr.exe Ignore NT AUTHORITY\LocalService 0			
Task Scheduler	Schedule	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			
Secondary Logon	seclogon	Running Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs Ignore LocalSystem 0			
System Event Notification	SENS	Running Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0			
Shell Hardware Detection	ShellHWDetection	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs Ignore LocalSystem 0			
Simple Mail Transfer Protocol (SMTP)	SMTPSVC	Running Auto	Share Process
c:\windows\system32\inetrv\inetinfo.exe Normal LocalSystem 0			
Print Spooler	Spooler	Stopped Manual	Own Process
c:\windows\system32\spoolsv.exe Normal LocalSystem 0			
SQLSERVERAGENT	SQLSERVERAGENT	Stopped Manual	Own Process
c:\sql80\mssql\binn\sqlagent.exe Normal LocalSystem 0			
Windows Image Acquisition (WIA)	stisvc	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k imgsvc Normal LocalSystem 0			
MS Software Snapshot Provider	SwPrv	Stopped Manual	Own Process
c:\windows\system32\dlhhost.exe /processid:{bd74b632-03d2-4df7-ac25-6442d88248e9} Ignore LocalSystem 0			
Performance Logs and Alerts	SysmonLog	Stopped Manual	Own Process
c:\windows\system32\smlogsvc.exe Normal NT AUTHORITY\NetworkService 0			
Telephony	TapiSrv	Stopped Manual	Share Process
c:\windows\system32\svchost.exe -k tapisrv Normal LocalSystem 0			
Terminal Services	TermService	Running Disabled	Share Process
c:\windows\system32\svchost.exe -k termsvcs Normal LocalSystem 0			

```

Themes Themes Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem
0
Telnet TlntSvr Stopped Manual Own Process
c:\windows\system32\tlntsvr.exe Normal LocalSystem 0

Distributed Link Tracking Server TrkSvr Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem
0
Distributed Link Tracking Client TrkWks Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem
0
Terminal Services Session Directory Tssdis Stopped Manual Own Process
c:\windows\system32\tssdis.exe Normal LocalSystem 0

Upload Manager uploadmgr Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem
0

Uninterruptible Power Supply UPS Stopped Manual Own Process
c:\windows\system32\ups.exe Normal NT AUTHORITY\LocalService
0

Volume Snapshot VSS Stopped Manual Own Process
c:\windows\system32\vssvc.exe Normal LocalSystem 0

Windows Time W32Time Running Auto Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem
0

SSL for World Wide Web Publishing W3SSL Running Auto Share Process
c:\windows\system32\svchost.exe -k w3ssl Normal LocalSystem
0

World Wide Web Publishing W3SVC Stopped Manual Share Process
c:\windows\system32\svchost.exe -k iissvcs Normal LocalSystem
0

WebClient WebClient Stopped Manual Share Process
c:\windows\system32\svchost.exe -k localservice Normal NT
AUTHORITY\LocalService 0

Windows Management Instrumentation winmgmt Running Auto Share Process
c:\windows\system32\svchost.exe -k netsvcs Ignore LocalSystem
0

Windows Management Instrumentation Driver Extensions Wmi Stopped Manual
Share Process c:\windows\system32\svchost.exe -k netsvcs Normal
LocalSystem 0

WMI Performance Adapter WmiApSrv Stopped Manual Own Process
c:\windows\system32\wbem\wmiapshr.exe Normal LocalSystem 0

Wireless Zero Configuration WZCSVC Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem
0

```

[Program Groups]

```

Group Name Name User Name
Accessories Default User:Accessories Default User
Accessories\Accessibility Default User:Accessories\Accessibility
Default User
Accessories\Entertainment Default User:Accessories\Entertainment
Default User
Startup Default User:Startup Default User
Accessories All Users:Accessories All Users

```

```

Accessories\Accessibility All Users:Accessories\Accessibility All
Users
Accessories\Communications All Users:Accessories\Communications All
Users
Accessories\Entertainment All Users:Accessories\Entertainment All
Users
Accessories\System Tools All Users:Accessories\System Tools All
Users
Administrative Tools All Users:Administrative Tools All Users

GigaNet All Users:GigaNet All Users
Intel VTune(TM) Performance Analyzer 6.0 All Users:Intel VTune(TM)
Performance Analyzer 6.0 All Users
Microsoft SQL Server All Users:Microsoft SQL Server All Users

```

```

Navisphere All Users:Navisphere All Users
Startup All Users:Startup All Users
Accessories NT AUTHORITY\SYSTEM:Accessories NT AUTHORITY\SYSTEM

```

```

Accessories\Accessibility NT
AUTHORITY\SYSTEM:Accessories\Accessibility NT AUTHORITY\SYSTEM
Accessories\Entertainment NT
AUTHORITY\SYSTEM:Accessories\Entertainment NT AUTHORITY\SYSTEM
Startup NT AUTHORITY\SYSTEM:Startup NT AUTHORITY\SYSTEM
Accessories MV-LE030P1\Administrator:Accessories MV-
LE030P1\Administrator
Accessories\Accessibility MV-
LE030P1\Administrator:Accessories\Accessibility MV-
LE030P1\Administrator
Accessories\Entertainment MV-
LE030P1\Administrator:Accessories\Entertainment MV-
LE030P1\Administrator
Administrative Tools MV-LE030P1\Administrator:Administrative Tools
MV-LE030P1\Administrator
Startup MV-LE030P1\Administrator:Startup MV-LE030P1\Administrator

```

[Startup Programs]

```

Program Command User Name Location
desktop desktop.ini NT AUTHORITY\SYSTEM Startup
desktop desktop.ini MV-LE030P1\Administrator Startup
desktop desktop.ini .DEFAULT Startup
desktop desktop.ini All Users Common Startup
IDW Logging Tool c:\windows\system32\idwlog.exe -3 All Users
Common Startup
UserFaultCheck %systemroot%\system32\dumprep 0 -u All Users
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

```

[OLE Registration]

```

Object Local Server
Sound (OLE2) sndrec32.exe
Media Clip mplay32.exe
Video Clip mplay32.exe /avi
MIDI Sequence mplay32.exe /mid
Sound Not Available
Media Clip Not Available

```

WordPad Document "%programfiles%\windows nt\accessories\wordpad.exe"

Bitmap Image mspaint.exe

[Windows Error Reporting]

Time Type Details

```

7/19/2001 4:33 PM Application Error Faulting application
NaviAgent.Exe, version 1.0.0.0, faulting module unknown, version 0.0.0.0,
fault address 0x00000000.&#x000d;&#x000a;
7/19/2001 8:48 AM Application Error Faulting application
Explorer.EXE, version 6.0.2496.0, faulting module SHELL32.dll, version
6.0.2496.0, fault address 0x001c44c3.&#x000d;&#x000a;

```

[Internet Settings]

[Internet Explorer]

[Following are sub-categories of this main category]
[Summary]

Item Value

[File Versions]

File	Version	Size	Date	Path	Company
------	---------	------	------	------	---------

[Connectivity]

Item Value

[Cache]

[Following are sub-categories of this main category]
[Summary]

Item Value

[List of Objects]

Program File	Status	CodeBase
--------------	--------	----------

[Content]

[Following are sub-categories of this main category]
[Summary]

Item Value

[Personal Certificates]

Issued To	Issued By	Validity	Signature	Algorithm
-----------	-----------	----------	-----------	-----------

[Other People Certificates]

Issued To	Issued By	Validity	Signature	Algorithm
-----------	-----------	----------	-----------	-----------

[Publishers]

[Following are sub-categories of this main category]
[Security]

Zone Security Level

Windows Datacenter Server Registry Information

Software\GigaNet

```

\registry\machine\software\GigaNet
  cLAN Management Console
    1.00.000
  GniConMgrMgmt2
    0
    BeginModify
      = 12:07:38
    Fabric
      Switch1
        SwitchModel = REG_DWORD 0x00001388
        SwitchRev = REG_DWORD 0x00002103
        Ports = n1 x n3 n4 n5 n6 n7 x
        SwitchEpromInfo = REG_BINARY 0x00000014 0x13880001
0x33300004 0x30373439 0x32303234 0x01000118
        LastModified
          = 12:07:38
        NodeList
          Node0
            Name = MV-LE030P1
            NICName = gnivia0
            Address = 0090FA001C0F
            MediaType =
            State = REG_DWORD 0x00000001
            SwitchPort = s1.1
            NodeNumber = REG_DWORD 0x00000001
          Node3
            Name = CLIENT1
            NICName = nic0
            Address = 0090FA001C8A
            MediaType =
            State = REG_DWORD 0x00000001
            SwitchPort = s1.3
            NodeNumber = REG_DWORD 0x00000003
          Node4
            Name = CLIENT5
            NICName = nic0
            Address = 0090FA001C1C
            MediaType =
            State = REG_DWORD 0x00000001
            SwitchPort = s1.4
            NodeNumber = REG_DWORD 0x00000004
          Node5
            Name = CLIENT4
            NICName = nic0
            Address = 0090FA001C35

```

```

        MediaType =
        State = REG_DWORD 0x00000001
        SwitchPort = s1.5
        NodeNumber = REG_DWORD 0x00000005
Node6
    Name = CLIENT3
    NICName = nic0
    Address = 0090FA00136E
    MediaType =
    State = REG_DWORD 0x00000001
    SwitchPort = s1.6
    NodeNumber = REG_DWORD 0x00000006
Node7
    Name = CLIENT2
    NICName = nic0
    Address = 0090FA001420
    MediaType =
    State = REG_DWORD 0x00000001
    SwitchPort = s1.7
    NodeNumber = REG_DWORD 0x00000007
2
BeginModify
    = 12:07:38
Fabric
    Switch1
        SwitchModel = REG_DWORD 0x00001388
        SwitchRev = REG_DWORD 0x00002103
        Ports = n1 n2 n3 n4 x n6 x x
        SwitchEpromInfo = REG_BINARY 0x00000014 0x13880001
0x33330004 0x34323539 0x38333135 0x01001014
    LastModified
        = 12:07:38
    NodeList
        Node0
            Name = MV-LE030P1
            NICName = gnivia2
            Address = 0090FA0014BC
            MediaType =
            State = REG_DWORD 0x00000001
            SwitchPort = s1.1
            NodeNumber = REG_DWORD 0x00000001
        Node2
            Name = CLIENT7
            NICName = nic0
            Address = 0090FA001C94
            MediaType =
            State = REG_DWORD 0x00000001
            SwitchPort = s1.2
            NodeNumber = REG_DWORD 0x00000002
        Node3
            Name = CLIENT8
            NICName = nic0
            Address = 0090FA000779
            MediaType =
            State = REG_DWORD 0x00000001
            SwitchPort = s1.3
            NodeNumber = REG_DWORD 0x00000003
        Node4
            Name = CLIENT6

```

```

        NICName = nic0
        Address = 0090FA001C7A
        MediaType =
        State = REG_DWORD 0x00000001
        SwitchPort = s1.4
        NodeNumber = REG_DWORD 0x00000004
Node6
    Name = CLIENT9
    NICName = nic0
    Address = 0090FA000E91
    MediaType =
    State = REG_DWORD 0x00000001
    SwitchPort = s1.6
    NodeNumber = REG_DWORD 0x00000006
LastModified
    = 12:07:37

```

Software\Microsoft\MSSQLServer

```

\registry\machine\software\Microsoft\MSSQLServer
Client
    SharedMemoryOn = REG_DWORD 0x00000001
    ConnectTo
        DSQUERY = DBNETLIB
    DB-Lib
        AutoAnsiToOem = ON
        UseIntlSettings = ON
    RPCNetlib
        LastConnect
            MV-LE030P1 = ncalrpc
    SuperSocketNetLib
        ProtocolOrder = REG_MULTI_SZ "np"
        Encrypt = REG_DWORD 0x00000000
        Np
            DefaultPipe = sql\query
        rpc
            RPCProtocols = REG_MULTI_SZ "nlrpc" \
                "nacn_np" \
                "nacn_ip_tcp" \
                "nacn_spx"
    Tcp
        DefaultPort = REG_DWORD 0x00000599
    TDS
        MV-LE030P2 = 7.0
        (local) = 7.0
        MV-LE030P1 = 7.0
        = 7.0
    MSSQLServer
        ListenOn = REG_MULTI_SZ "SSMSSH70" \
            "SSNETLIB"
        ResourceMgrID = {634D63E8-844F-4852-AD22-6EA4ECCF914F}
        SetHostName = REG_DWORD 0x00000000
        AuditLevel = REG_DWORD 0x00000000
        LoginMode = REG_DWORD 0x00000002
        Tapeloadwaittime = REG_DWORD 0xffffffff
        DefaultLogin = guest
        Map_ = \

```

```

Map# = -
Map$ =
BackupDirectory = c:\sql80\MSSQL\BACKUP
DefaultDomain = MV-LE030P2
DefaultCollationName = Latin1_General_BIN
CurrentVersion
    checksum = REG_BINARY 0x000000a1 0x32323937 0x38353163
0x64376561 0x37646334 0x61643035 0x62343330 0x32636437 0x34306630
0x36303633 0x33383066 0x36666463 0x31363834 0x33613431 0x34633139
0x38326333 0x61383036 0x36646166 \
    0x63666462 0x66343864 0x63316134 0x34323632
0x61646162 0x61613636 0x37393734 0x61336663 0x37333035 0x32323363
0x31353231 0x39613462 0x62623165 0x64303630 0x64323265 0x32346362
0x62353261 0x64643066 0x62326637 \
    0x32616135 0x62663461 0x63613138 0x38326230
0x00000000
    RegisteredOwner = MVSAMM
    SerialNumber = REG_DWORD 0x81d00040
    CurrentVersion = 8.00.391
    Language = REG_DWORD 0x00000409
Parameters
    SQLArg0 = -dc:\sql80\MSSQL\data\master.mdf
    SQLArg1 = -ec:\sql80\MSSQL\log\ERRORLOG
    SQLArg2 = -lc:\sql80\MSSQL\data\mastlog.ldf
RPCNetLib
    Security =
SuperSocketNetLib
    ProtocolList = REG_MULTI_SZ "np" \
        "via"
    Encrypt = REG_DWORD 0x00000000
Np
    PipeName =
rpc
    RPCProtocols = REG_MULTI_SZ
    Encrypt = REG_DWORD 0x00000000
Tcp
    TcpHideFlag = REG_DWORD 0x00000000
    TcpPort =
    TcpDynamicPorts =
VIA
    Vendor = Gigaset
    ListenInfo =
2:1421[0xffffffff],2:1422[0xffffffff],2:1423[0xffffffff],2:1424[0xffffffff
],0:1431[0xffffffff],0:1432[0xffffffff],0:1433[0xffffffff],0:1434[0xfffff
ff],0:1435[0xffffffff]
    RecognizedVendors = REG_MULTI_SZ "Gigaset"
Providers
    AllowInProcess = REG_DWORD 0x00000001
ADSDSOObject
    AllowInProcess = REG_DWORD 0x00000001
DB2OLEDB
    AllowInProcess = REG_DWORD 0x00000001
Microsoft.Jet.OLEDB.4.0
    AllowInProcess = REG_DWORD 0x00000001
MSDAORA
    AllowInProcess = REG_DWORD 0x00000001
MSDASQL
    AllowInProcess = REG_DWORD 0x00000001
MSIDX

```

```

    AllowInProcess = REG_DWORD 0x00000001
MSQLImpProv
    AllowInProcess = REG_DWORD 0x00000001
MSSEARCHSQL
    AllowInProcess = REG_DWORD 0x00000001
SQLOLEDB
    AllowInProcess = REG_DWORD 0x00000001
    DisallowAdhocAccess = REG_DWORD 0x00000000
Setup
    SQLPath = c:\sql80\MSSQL
    SQLDataRoot = c:\sql80\MSSQL
    SourcePath = D:
    firststart = REG_DWORD 0x00000000
SQLServerAgent
    JobHistoryMaxRowsPerJob = REG_DWORD 0x00000064
    RestartServer = REG_DWORD 0x00000001
    DownloadedMaxRows = REG_DWORD 0x00000064
    JobHistoryMaxRows = REG_DWORD 0x000003e8
    ErrorLoggingLevel = REG_DWORD 0x00000003
    NonAlertableErrors = 1204,4002
    MSXServerName =
    ServerHost =
    WorkingDirectory = c:\sql80\MSSQL\JOBS
    ErrorLogFile = c:\sql80\MSSQL\LOG\SQLAGENT.OUT
    SysAdminOnly = REG_DWORD 0x00000001
Subsystems
    CmdExec =
c:\sql80\MSSQL\BINN\SQLCMDSS.DLL,NULL,CmdExecStart,CmdEvent,CmdExecStop,10
    Snapshot = c:\sql80\MSSQL\BINN\SQLREPSS.DLL,C:\Program
Files\Microsoft SQL
Server\80\COM\SNAPSHOT.EXE,ReplStart,ReplEvent,ReplStop,100
    Distribution = c:\sql80\MSSQL\BINN\SQLREPSS.DLL,C:\Program
Files\Microsoft SQL
Server\80\COM\DISTRIB.EXE,ReplStart,ReplEvent,ReplStop,100
    Merge = c:\sql80\MSSQL\BINN\SQLREPSS.DLL,C:\Program
Files\Microsoft SQL
Server\80\COM\REPLMERG.EXE,ReplStart,ReplEvent,ReplStop,100
    ActiveScripting =
c:\sql80\MSSQL\BINN\SQLATXSS.DLL,NULL,ActiveScriptStart,ActiveScriptEvent,
ActiveScriptStop,10
    QueueReader = c:\sql80\MSSQL\BINN\SQLREPSS.DLL,C:\Program
Files\Microsoft SQL
Server\80\COM\QDRSVC.EXE,ReplStart,ReplEvent,ReplStop,100
    LogReader = c:\sql80\MSSQL\BINN\SQLREPSS.DLL,C:\Program
Files\Microsoft SQL
Server\80\COM\LOGREAD.EXE,ReplStart,ReplEvent,ReplStop,25
Tracking
    {E07FDDA4-5A21-11d2-9DAD-00C04F79D434} =
    {6DC86044-0C71-11d3-9E18-00C04F79D434} =
    {E07FDDA8-5A21-11d2-9DAD-00C04F79D434} =
    {E07FDDA9-5A21-11d2-9DAD-00C04F79D434} =
    {E07FDDAA-5A21-11d2-9DAD-00C04F79D434} =
    {E07FDDAC-5A21-11d2-9DAD-00C04F79D434} =
    {E07FDDAF-5A21-11d2-9DAD-00C04F79D434} =
    {E07FDDBE-5A21-11d2-9DAD-00C04F79D434} =
    {E07FDDC0-5A21-11d2-9DAD-00C04F79D434} =
    {E07FDDBF-5A21-11d2-9DAD-00C04F79D434} =
Shortcuts
    Enterprise Manager =

```

```
Query Analyzer =
Profiler =
Client Network Utility =
Configure SQL XML Support in IIS =
Import and Export Data =
Service Manager =
Server Network Utility =
Books Online =
```

Software\Microsoft\Microsoft SQL Server

```
\registry\machine\software\Microsoft\Microsoft SQL Server
  SsrpActiveServer = (***) MISSING TRAILING NULL CHARACTER (***)
  InstalledInstances = REG_MULTI_SZ "MSSQLSERVER"
  SqlMdacRegRefCount = REG_DWORD 0x00000001
  8.00.000
  80
  MSSQLLicenseInfo
    MSSQL8.00
    FlipAllow = REG_DWORD 0x00000001
    ConcurrentLimit = REG_DWORD 0x0000000a
    DisplayName = SQL Server 2000
    FamilyDisplayName = Microsoft SQL Server
    Mode = REG_DWORD 0x00000000
  Registration
    DigitalProductID = REG_BINARY 0x000000a4 0x000000a4 0x00000003
0x33393335 0x30302d34 0x30302d30 0x39303030 0x35302d35 0x00353130
0x0000000a 0x2d303138 0x36353030 0x00000030 0x00000000 0x00000012
0x0f0aa600 0xef9e9d5e \
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x32353830 0x00000031 0x00000000 0x000000de3 0xbc2bde3e 0x00000800
0x00001561 0x00000000 \
0x00000000 0x00000000 0x00000000 0x00000000 0x00000000
0x00000000 0x00000000 0x56f73a61
  ProductID = 53934-000-0000095-05015
  CD_KEY = BC6JV-WVYQ9-PC8H6-W938R-8BWJD
  Replication
    MergeReplicationProvider
      MsJet
      = {f159cf30-0db4-11d1-b272-00aa00b8de95}
  Tools
    ClientSetup
      SQLPath = C:\Program Files\Microsoft SQL Server\80\Tools
      CurrentVersion
        CurrentVersion = 8.00.391
        Language = REG_DWORD 0x00000409
    Help
      RefCount = REG_DWORD 0x00000001
      HelpFile = C:\Program Files\Microsoft SQL
Server\80\Tools\Books\SQL80.col
      HelpLCID = 1033
      HelpPath = C:\Program Files\Microsoft SQL
Server\80\Tools\Books
      Service Manager
      Action Verify = REG_DWORD 0x00000000
```

```
Default = MSSQLSERVER
Remote = REG_DWORD 0x00000001
Services = REG_MULTI_SZ "MSDTC"
SQLEW
  CustomizableAlertBaseURL =
  CustomizableAlertDefaultButtonText = &Send Feedback
  CustomizableAlertDefaultBtnShortText = Feedback&k
  Replication
    PerfmonFile = C:\Program Files\Microsoft SQL
Server\80\Tools\BINN\REPLMON.PMC
    Wizards
      Web Assistant = C:\Program Files\Microsoft SQL
Server\80\Tools\BINN\semwebwz.DLL^WebWizardEntry
    Tracking
      {E07FDDA8-5A21-11d2-9DAD-00C04F79D434} = REG_DWORD
0x00000001
      {E07FDDA9-5A21-11d2-9DAD-00C04F79D434} = REG_DWORD
0x00000001
      {E07FDDAA-5A21-11d2-9DAD-00C04F79D434} = REG_DWORD
0x00000001
      {E07FDDAC-5A21-11d2-9DAD-00C04F79D434} = REG_DWORD
0x00000001
      {E07FDDAF-5A21-11d2-9DAD-00C04F79D434} = REG_DWORD
0x00000001
      {E07FDDBE-5A21-11d2-9DAD-00C04F79D434} = REG_DWORD
0x00000001
      {E07FDDBF-5A21-11d2-9DAD-00C04F79D434} = REG_DWORD
0x00000001
  Shortcuts
    Enterprise Manager = REG_DWORD 0x00000001
    Query Analyzer = REG_DWORD 0x00000001
    Profiler = REG_DWORD 0x00000001
    Client Network Utility = REG_DWORD 0x00000001
    Configure SQL XML Support in IIS = REG_DWORD
0x00000001
    Import and Export Data = REG_DWORD 0x00000001
    Service Manager = REG_DWORD 0x00000001
    Server Network Utility = REG_DWORD 0x00000001
    Books Online = REG_DWORD 0x00000001
CurrentControlSet\Services\dasd
\registry\machine\system\CurrentControlSet\Services\dasd
  Type = REG_DWORD 0x00000001
  Start = REG_DWORD 0x00000000
  ErrorControl = REG_DWORD 0x00000001
  ImagePath = REG_EXPAND_SZ System32\DRIVERS\dasd.sys
  DisplayName = Unisys DASD Driver v1.3.15
  Parameters
    LogLevel = REG_DWORD 0x00000002
    Direct = REG_DWORD 0x00000001
    Path = C:\WINDOWS
  Security [17 1]
    Security = REG_BINARY 0x000000a8 0x80140001 0x00000090 0x0000009c
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
```



```

64 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&34cb3ec0&0&DASD00050010
65 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&34cb3ec0&0&DASD00050011
66 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&34cb3ec0&0&DASD00050016
67 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&34cb3ec0&0&DASD00050017
68 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060000
69 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060001
70 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060006
71 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060007
72 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060008
73 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060009
74 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD0006000e
75 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD0006000f
76 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060010
77 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060011
78 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060016
79 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&1155b2b4&0&DASD00060017
80 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070000
81 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070001
82 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070006
83 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070007
84 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070008
85 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070009
86 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD0007000e
87 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD0007000f
88 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070010
89 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070011
90 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070016
91 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_0___\5&187f1edc&0&DASD00070017
92 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_10___\5&14b7c5a0&0&DASD00080000

```

```

93 =
SCSI\Disk&Ven_DGC&Prod_RAID&Rev_10___\5&14b7c5a0&0&DASD00080003

```

CurrentControlSet\Services\GniConMgr

```

\registry\machine\system\CurrentControlSet\Services\GniConMgr
  Type = REG_DWORD 0x00000010
  Start = REG_DWORD 0x00000002
  ErrorControl = REG_DWORD 0x00000001
  ImagePath = REG_EXPAND_SZ %SystemRoot%\System32\GnConMgr.exe
  DisplayName = cLAN Connection Manager
  Group = TDI
  DependOnService = REG_MULTI_SZ "GniVIA"
  DependOnGroup = REG_MULTI_SZ
  ObjectName = LocalSystem
  Parameters
    NicName = GniVIA
  Security [17 1]
    Security = REG_BINARY 0x000000a8 0x80140001 0x00000090 0x0000009c
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
0x00000101 0x01000000 0x00000000 0x00600002 0x00000004 0x00140000
0x000201fd 0x00000101 0x05000000 \
    0x00000012 0x00180000 0x000f01ff 0x00000201 0x05000000
0x00000020 0x00000220 0x00140000 0x0002018d 0x00000101 0x05000000
0x0000000b 0x00180000 0x000201fd 0x00000201 0x05000000 0x00000020
0x00000223 0x00000101 0x05000000 \
    0x00000012 0x00000101 0x05000000 0x00000012
  Enum
    0 = Root\LEGACY_GNICONMGR\0000
  Count = REG_DWORD 0x00000001
  NextInstance = REG_DWORD 0x00000001

```

CurrentControlSet\Services\GNINDIS

```

\registry\machine\system\CurrentControlSet\Services\GNINDIS
  Type = REG_DWORD 0x00000001
  Start = REG_DWORD 0x00000002
  ErrorControl = REG_DWORD 0x00000001
  Tag = REG_DWORD 0x0000000d
  ImagePath = REG_EXPAND_SZ System32\DRIVERS\gnindis.sys
  DisplayName = cLAN NDIS Driver
  Group = NDIS
  Security [17 1]
    Security = REG_BINARY 0x000000a8 0x80140001 0x00000090 0x0000009c
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
0x00000101 0x01000000 0x00000000 0x00600002 0x00000004 0x00140000
0x000201fd 0x00000101 0x05000000 \
    0x00000012 0x00180000 0x000f01ff 0x00000201 0x05000000
0x00000020 0x00000220 0x00140000 0x0002018d 0x00000101 0x05000000
0x0000000b 0x00180000 0x000201fd 0x00000201 0x05000000 0x00000020
0x00000223 0x00000101 0x05000000 \
    0x00000012 0x00000101 0x05000000 0x00000012
  Enum

```

```
Count = REG_DWORD 0x00000002
NextInstance = REG_DWORD 0x00000002
0 = PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&1b2f0ce&0&08
1 = PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&a985f74&0&08
```

CurrentControlSet\Services\GniVIA

```
\registry\machine\system\CurrentControlSet\Services\GniVIA
Type = REG_DWORD 0x00000001
Start = REG_DWORD 0x00000002
ErrorControl = REG_DWORD 0x00000001
Tag = REG_DWORD 0x00000007
ImagePath = REG_EXPAND_SZ System32\DRIVERS\GniVIA.sys
DisplayName = cLAN VIA Driver
Group = PNP_TDI
Linkage
  Bind = REG_MULTI_SZ "\Device\{2CFD9D8A-12C1-4C59-9FF9-
A8B326C34B3D}" \
"\Device\{C7084E8E-79DF-4B41-BA2C-
04D6C1039C36}" \
"\Device\{B273422F-30FF-4F86-B6A8-
EEECD31E8F37}" \
"\Device\{87DA539C-4C8C-41D8-861C-
11BD6DC1EBD9}" \
"\Device\{C52D958A-FCD8-419F-8F1F-
C49AD99857A8}"
  Route = REG_MULTI_SZ "\{2CFD9D8A-12C1-4C59-9FF9-A8B326C34B3D}" \
"\{C7084E8E-79DF-4B41-BA2C-04D6C1039C36}" \
"\{B273422F-30FF-4F86-B6A8-EEECD31E8F37}" \
"\{87DA539C-4C8C-41D8-861C-11BD6DC1EBD9}" \
"\{C52D958A-FCD8-419F-8F1F-C49AD99857A8}" \
Export = REG_MULTI_SZ "\Device\GniVIA_{2CFD9D8A-12C1-4C59-9FF9-
A8B326C34B3D}" \
"\Device\GniVIA_{C7084E8E-79DF-4B41-BA2C-
04D6C1039C36}" \
"\Device\GniVIA_{B273422F-30FF-4F86-B6A8-
EEECD31E8F37}" \
"\Device\GniVIA_{87DA539C-4C8C-41D8-861C-
11BD6DC1EBD9}" \
"\Device\GniVIA_{C52D958A-FCD8-419F-8F1F-
C49AD99857A8}"
  Parameters
  =
  MaxNumberNics = REG_DWORD 0x00000005
Performance
  Open = OpenNicPerformanceData
  Close = CloseNicPerformanceData
  Collect = CollectNicPerformanceData
  Library = gni_perf.dll
  WbemAdapFileSignature = REG_BINARY 0x00000010 0xbb76a098
0x6405bf8f 0xaa9f11a4 0x7eea5948
  WbemAdapFileTime = REG_BINARY 0x00000008 0x48297f74 0x01c0f9d9
```

```
WbemAdapFileSize = REG_DWORD 0x0000c000
WbemAdapStatus = REG_DWORD 0x00000000
Last Counter = REG_DWORD 0x00000df4
Last Help = REG_DWORD 0x00000df5
First Counter = REG_DWORD 0x00000db6
First Help = REG_DWORD 0x00000db7
```

Security [17 1]

```
Security = REG_BINARY 0x000000a8 0x80140001 0x00000090 0x0000009c
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
0x00000101 0x01000000 0x00000000 0x00600002 0x00000004 0x00140000
0x000201fd 0x00000101 0x05000000 \
0x00000012 0x00180000 0x000f01ff 0x00000201 0x05000000
0x00000020 0x00000220 0x00140000 0x0002018d 0x00000101 0x05000000
0x0000000b 0x00180000 0x000201fd 0x00000201 0x05000000 0x00000020
0x00000223 0x00000101 0x05000000 \
0x00000012 0x00000101 0x05000000 0x00000012
```

Enum

```
0 = Root\LEGACY_GNIVIA\0000
Count = REG_DWORD 0x00000001
NextInstance = REG_DWORD 0x00000001
```

CurrentControlSet\Services\i2oubbs

```
\registry\machine\system\CurrentControlSet\Services\i2oubbs
Type = REG_DWORD 0x00000001
Start = REG_DWORD 0x00000000
ErrorControl = REG_DWORD 0x00000001
ImagePath = REG_EXPAND_SZ System32\DRIVERS\i2oubbs.sys
DisplayName = Unisys Fibre Channel Driver v1.3.15
Parameters
  LogLevel = REG_DWORD 0x00000002
  Path = C:\WINDOWS
  PortDriver = REG_DWORD 0x00000001
  VA = REG_DWORD 0x00000001
  PostListBatchSize = REG_DWORD 0x00000001
  IIDelay = REG_DWORD 0x00000200
  PostListBatchDelayMax = REG_DWORD 0x00000040
  DpcPri = REG_DWORD 0x00000002
  VA_More = REG_DWORD 0x00000000
  XP = REG_DWORD 0x00000001
  GenDisk = REG_DWORD 0x00000000
  ScanReg = REG_DWORD 0x00000000
  nopStartStopUnit = REG_DWORD 0x00000000
  Download = REG_DWORD 0x00000000
  DisablePullMessages = REG_DWORD 0x00000000
  DisableOutboundOption = REG_DWORD 0x00000000
  DcpPri = REG_DWORD 0x00000001
  SepIIDelay = REG_DWORD 0x00000020
  SepBusNum = REG_DWORD 0x00000008
  ProcessorGroupMask = REG_DWORD 0x00000000
Security [17 1]
  Security = REG_BINARY 0x000000a8 0x80140001 0x00000090 0x0000009c
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
0x00000101 0x01000000 0x00000000 0x00600002 0x00000004 0x00140000
0x000201fd 0x00000101 0x05000000 \
```

```
0x00000012 0x00180000 0x000f01ff 0x00000201 0x05000000
0x00000020 0x00000220 0x00140000 0x0002018d 0x00000101 0x05000000
0x0000000b 0x00180000 0x000201fd 0x00000201 0x05000000 0x00000020
0x00000223 0x00000101 0x05000000 \
0x00000012 0x00000101 0x05000000 0x00000012
```

Enum

```
Count = REG_DWORD 0x00000009
NextInstance = REG_DWORD 0x00000009
0 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&6ca15c9&0&4010
1 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&6afa766&0&4008
2 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&22f87bf1&0&4008
3 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&de6f207&0&4008
4 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&3d39d8e&0&4000
5 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&19ad16c8&0&4008
6 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&2ec99f52&0&4000
7 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&ad6b475&0&4000
8 = PCI\VEN_1011&DEV_1065&SUBSYS_C0FC1018&REV_04\4&39369b90&0&4008
```

CurrentControlSet\Services\MSSQLSERVER

```
\registry\machine\system\CurrentControlSet\Services\MSSQLSERVER
```

```
Type = REG_DWORD 0x00000010
Start = REG_DWORD 0x00000003
ErrorControl = REG_DWORD 0x00000001
ImagePath = REG_EXPAND_SZ c:\sql80\MSSQL\bin\sqlservr.exe
DisplayName = MSSQLSERVER
ObjectName = LocalSystem
Linkage
```

```
Export = REG_MULTI_SZ "MSSQLSERVER"
```

Performance

```
Library = c:\sql80\MSSQL\BINN\SQLCTR80.DLL
Collect = CollectSQLPerformanceData
Open = OpenSQLPerformanceData
Close = CloseSQLPerformanceData
Last Counter = REG_DWORD 0x00000d68
Last Help = REG_DWORD 0x00000d69
First Counter = REG_DWORD 0x00000c54
First Help = REG_DWORD 0x00000c55
```

```
WbemAdapFileSignature = REG_BINARY 0x00000010 0xc1b9a0d6
0x637885d1 0xf28b314f 0x5d1a5118
```

```
WbemAdapFileTime = REG_BINARY 0x00000008 0xe27ae5a6 0x01c0f9d7
WbemAdapFileSize = REG_DWORD 0x0000803b
WbemAdapStatus = REG_DWORD 0x00000000
```

Security [17 1]

```
Security = REG_BINARY 0x000000a8 0x80140001 0x00000090 0x0000009c
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
0x00000101 0x01000000 0x00000000 0x00600002 0x00000004 0x00140000
0x000201fd 0x00000101 0x05000000 \
0x00000012 0x00180000 0x000f01ff 0x00000201 0x05000000
0x00000020 0x00000220 0x00140000 0x0002018d 0x00000101 0x05000000
0x0000000b 0x00180000 0x000201fd 0x00000201 0x05000000 0x00000020
0x00000223 0x00000101 0x05000000 \
0x00000012 0x00000101 0x05000000 0x00000012
```

Enum

```
0 = Root\LEGACY_MSSQLSERVER\0000
Count = REG_DWORD 0x00000001
```

```
NextInstance = REG_DWORD 0x00000001
```

CurrentControlSet\Services\SQLSERVERAGENT

```
\registry\machine\system\CurrentControlSet\Services\SQLSERVERAGENT
```

```
Type = REG_DWORD 0x00000010
Start = REG_DWORD 0x00000003
ErrorControl = REG_DWORD 0x00000001
ImagePath = REG_EXPAND_SZ c:\sql80\MSSQL\bin\sqlagent.exe
DisplayName = SQLSERVERAGENT
DependOnService = REG_MULTI_SZ "MSSQLSERVER"
DependOnGroup = REG_MULTI_SZ
ObjectName = LocalSystem
Security [17 1]
```

```
Security = REG_BINARY 0x000000a8 0x80140001 0x00000090 0x0000009c
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
0x00000101 0x01000000 0x00000000 0x00600002 0x00000004 0x00140000
0x000201fd 0x00000101 0x05000000 \
0x00000012 0x00180000 0x000f01ff 0x00000201 0x05000000
0x00000020 0x00000220 0x00140000 0x0002018d 0x00000101 0x05000000
0x0000000b 0x00180000 0x000201fd 0x00000201 0x05000000 0x00000020
0x00000223 0x00000101 0x05000000 \
0x00000012 0x00000101 0x05000000 0x00000012
```

Windows 2000 Server Client Configuration Information

(common to 9 clients except for System Name, IP Address & MAC Address)

System Information report written at: 07/20/2001 02:25:18 PM
client1[System Information]

[Following are sub-categories of this main category]

[System Summary]

Item	Value
OS Name	Microsoft Windows 2000 Server
Version	5.0.2195 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	CLIENT1
System Manufacturer	IBM
System Model	Unisys e-action Enterprise Server
System Type	X86-based PC
Processor	x86 Family 6 Model 8 Stepping 6 GenuineIntel ~1000 Mhz
Processor	x86 Family 6 Model 8 Stepping 6 GenuineIntel ~1000 Mhz
BIOS Version	IBM BIOS Ver 7.0
Windows Directory	C:\WINNT
System Directory	C:\WINNT\System32
Boot Device	\Device\Harddisk0\Partition1
Locale	United States
User Name	CLIENT1\Administrator
Time Zone	Pacific Daylight Time
Total Physical Memory	523,796 KB
Available Physical Memory	449,948 KB

Total Virtual Memory 1,802,300 KB
Available Virtual Memory 1,673,108 KB
Page File Space 1,278,504 KB
Page File C:\pagefile.sys

[Hardware Resources]

[Following are sub-categories of this main category]

[Conflicts/Sharing]

Resource Device
No conflicted/shared resources

[DMA]

Channel	Device	Status
2	Standard floppy disk controller	OK
4	Direct memory access controller	OK

[Forced Hardware]

Device PNP Device ID
No Forced Hardware

[I/O]

Address Range	Device	Status
0x0000-0x0CF7	PCI bus	OK
0x0000-0x0CF7	Direct memory access controller	OK
0x0D00-0x207F	PCI bus	OK
0xFD00-0xFFFF	PCI bus	OK
0xFD00-0xFFFF	Motherboard resources	OK
0x03B0-0x03BB	S3 Inc. Savage4	OK
0x03C0-0x03DF	S3 Inc. Savage4	OK
0x2000-0x201F	IBM Netfinity Fault Tolerance PCI Adapter	OK
0x2040-0x207F	Intel(R) PRO/100+ Server Adapter (PILA8470B)	OK
0x0A79-0x0A79	ISAPNP Read Data Port	OK
0x0279-0x0279	ISAPNP Read Data Port	OK
0x02F4-0x02F7	ISAPNP Read Data Port	OK
0x0060-0x0060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x0064-0x0064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x03F0-0x03F5	Standard floppy disk controller	OK
0x03F7-0x03F7	Standard floppy disk controller	OK
0x0378-0x037F	Printer Port (LPT1)	OK
0x03F8-0x03FF	Communications Port (COM1)	OK
0x02F8-0x02FF	Communications Port (COM2)	OK
0x0020-0x0021	Advanced programmable interrupt controller	OK
0x00A0-0x00A1	Advanced programmable interrupt controller	OK
0x04D0-0x04D1	Advanced programmable interrupt controller	OK
0x0080-0x008F	Direct memory access controller	OK
0x00C0-0x00DF	Direct memory access controller	OK
0x0040-0x0043	System timer	OK
0x0070-0x0073	System CMOS/real time clock	OK
0x0061-0x0061	System speaker	OK
0x00F0-0x00FF	Numeric data processor	OK
0x0F50-0x0F58	Motherboard resources	OK

0xFE80-0xFE8F	Motherboard resources	OK
0xFEC0-0xFEDF	Motherboard resources	OK
0xEE9B-0xEE9B	Motherboard resources	OK
0x00E8-0x00E9	Not Available	OK
0x0840-0x084F	Standard Dual Channel PCI IDE Controller	OK
0x01F0-0x01F7	Primary IDE Channel	OK
0x03F6-0x03F6	Primary IDE Channel	OK
0x0170-0x0177	Secondary IDE Channel	OK
0x0376-0x0376	Secondary IDE Channel	OK
0x2080-0x208F	PCI bus	OK
0x2100-0x21FF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0x2200-0x22FF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK

[IRQs]

IRQ Number	Device
30	Microsoft ACPI-Compliant System
31	S3 Inc. Savage4
27	IBM Netfinity Fault Tolerance PCI Adapter
18	Intel(R) PRO/100+ Server Adapter (PILA8470B)
1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12	PS/2 Compatible Mouse
6	Standard floppy disk controller
4	Communications Port (COM1)
3	Communications Port (COM2)
8	System CMOS/real time clock
13	Numeric data processor
5	Not Available
14	Primary IDE Channel
11	Standard OpenHCD USB Host Controller
28	Adaptec AIC-7899 Ultra160/m PCI SCSI Card
29	Adaptec AIC-7899 Ultra160/m PCI SCSI Card
20	cLAN Host Adapter

[Memory]

Range	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	S3 Inc. Savage4	OK
0xF0000000-0xFFFFFFFF	PCI bus	OK
0xF0000000-0xFFFFFFFF	S3 Inc. Savage4	OK
0xFEB80000-0xFEBFFFFFF	S3 Inc. Savage4	OK
0xFEB7FC00-0xFEB7FC1F	IBM Netfinity Fault Tolerance PCI Adapter	OK
0xFEB7E000-0xFEB7EFFF	Intel(R) PRO/100+ Server Adapter (PILA8470B)	OK
0xFEA00000-0xFEAFFFFFF	Intel(R) PRO/100+ Server Adapter (PILA8470B)	OK
0xFEC00000-0xFECFFFFFF	Advanced programmable interrupt controller	OK
0xFEE00000-0xFEEFFFFFF	Advanced programmable interrupt controller	OK
0xFF700000-0xFF70FFFF	Standard OpenHCD USB Host Controller	OK
0x20000000-0xEFFFFFFF	PCI bus	OK
0xEFFF0000-0xEFFFFFFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0xEFFF0000-0xEFFFFFFF	Adaptec AIC-7899 Ultra160/m PCI SCSI Card	OK
0xEFFC0000-0xEFFDFFFF	cLAN Host Adapter	OK
0xEFC00000-0xEFDFFFFF	cLAN Host Adapter	OK
0xEE000000-0xEEFFFFFF	cLAN Host Adapter	OK
0xEFFE0000-0xEFFEFFFF	cLAN Host Adapter	OK
0x0000-0x9FFF	System board	OK
0x100000-0xFFFFFFFF	System board	OK
0x1000000-0x1FFFFFFF	System board	OK
0xE0000-0xFFFFF	System board	OK

```

0xFFFE0000-0xFFFFFFFF System board OK
0xC8000-0xCBFFF System board OK
0xCC000-0xCFFFF System board OK
0xD0000-0xD3FFF System board OK

```

[Components]

[Following are sub-categories of this main category]

[Multimedia]

[Following are sub-categories of this main category]

[Audio Codecs]

Codec	Manufacturer	Description	Status	File	Version	Size
		Creation Date				
c:\winnt\system32\msg723.acm	Microsoft Corporation		OK			
		C:\WINNT\System32\MSG723.ACM	4.4.3385		106.77	KB (109,328 bytes)
		1/16/2001 2:59:40 PM				
c:\winnt\system32\iac25_32.ax	Intel Corporation		Indeo® audio software			
		OK C:\WINNT\System32\IAC25_32.AX	2.05.53		195.00	KB (199,680 bytes)
		12/7/1999 4:00:00 AM				
c:\winnt\system32\tssoft32.acm	DSP GROUP, INC.		OK			
		C:\WINNT\System32\TSSOFT32.ACM	1.01		9.27	KB (9,488 bytes)
		12/7/1999 4:00:00 AM				
c:\winnt\system32\msg711.acm	Microsoft Corporation		OK			
		C:\WINNT\System32\MSG711.ACM	5.00.2134.1		10.27	KB (10,512 bytes)
		12/7/1999 4:00:00 AM				
c:\winnt\system32\imaadp32.acm	Microsoft Corporation		OK			
		C:\WINNT\System32\IMAADP32.ACM	5.00.2134.1		16.27	KB (16,656 bytes)
		12/7/1999 4:00:00 AM				
c:\winnt\system32\msadp32.acm	Microsoft Corporation		OK			
		C:\WINNT\System32\MSADP32.ACM	5.00.2134.1		14.77	KB (15,120 bytes)
		12/7/1999 4:00:00 AM				
c:\winnt\system32\msgsm32.acm	Microsoft Corporation		OK			
		C:\WINNT\System32\MSGSM32.ACM	5.00.2134.1		22.27	KB (22,800 bytes)
		12/7/1999 4:00:00 AM				
c:\winnt\system32\lhacm.acm	Microsoft Corporation		OK			
		C:\WINNT\System32\LHACM.ACM	4.4.3385		33.27	KB (34,064 bytes)
		1/16/2001 2:59:41 PM				

[Video Codecs]

Codec	Manufacturer	Description	Status	File	Version	Size
		Creation Date				
c:\winnt\system32\ir50_32.dll	Intel Corporation		Indeo® video		5.10	
		OK C:\WINNT\System32\IR50_32.DLL	R.5.10.15.2.55		737.50	KB (755,200 bytes)
		12/7/1999 4:00:00 AM				
c:\winnt\system32\msh261.drv	Microsoft Corporation		OK			
		C:\WINNT\System32\MSH261.DRV	4.4.3385		163.77	KB (167,696 bytes)
		1/16/2001 2:59:40 PM				
c:\winnt\system32\msh263.drv	Microsoft Corporation		OK			
		C:\WINNT\System32\MSH263.DRV	4.4.3385		252.27	KB (258,320 bytes)
		1/16/2001 2:59:16 PM				
c:\winnt\system32\iccvid.dll	Radius Inc.		OK			
		C:\WINNT\System32\ICCVID.DLL	1.10.0.6		108.00	KB (110,592 bytes)
		12/7/1999 4:00:00 AM				

```

c:\winnt\system32\msvidc32.dll Microsoft Corporation OK
C:\WINNT\System32\MSVIDC32.DLL 5.00.2134.1 27.27 KB
(27,920 bytes) 12/7/1999 4:00:00 AM
c:\winnt\system32\msrle32.dll Microsoft Corporation OK
C:\WINNT\System32\MSRLE32.DLL 5.00.2134.1 10.77 KB (11,024
bytes) 12/7/1999 4:00:00 AM
c:\winnt\system32\ir32_32.dll Intel(R) Corporation OK
C:\WINNT\System32\IR32_32.DLL Not Available 194.50 KB (199,168
bytes) 12/7/1999 4:00:00 AM

```

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	False
Media Type	CD-ROM
Name	LITEON CD-ROM LTN403S
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMLITEON_CD-ROM_LTN403S_____QU05____\5&326853DD&0&0.0.0

[Sound Device]

Item	Value
No sound devices	

[Display]

Item	Value
Name	S3 Inc. Savage4
PNP Device ID	PCI\VEN_5333&DEV_8A22&SUBSYS_01C51014&REV_04\3&267A616A&0&08
Adapter Type	S3 Savage4, S3 compatible
Adapter Description	S3 Inc. Savage4
Adapter RAM	8.00 MB (8,388,608 bytes)
Installed Drivers	s3sav4.dll
Driver Version	5.01.840.0001
INF File	s3sav4.inf (S3Inc section)
Color Planes	1
Color Table Entries	256
Resolution	800 x 600 x 60 hertz
Bits/Pixel	8

[Infrared]

Item	Value
No infrared devices	

[Input]

[Following are sub-categories of this main category]

[Keyboard]

Item Value
Description Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Name Enhanced (101- or 102-key)
Layout 00000409
PNP Device ID ACPI\PNP0303\4&F0B8F99&0
NumberOfFunctionKeys 12

[Pointing Device]

Item Value
Hardware Type PS/2 Compatible Mouse
Number of Buttons 2
Status OK
PNP Device ID ACPI\PNP0F13\4&F0B8F99&0
Power Management Supported False
Double Click Threshold 6
Handedness Right Handed Operation

[Modem]

Item Value
No modems

[Network]

[Following are sub-categories of this main category]

[Adapter]

Item Value
Name [00000000] IBM Netfinity Fault Tolerance PCI Adapter
Adapter Type Ethernet 802.3
Product Name IBM Netfinity Fault Tolerance PCI Adapter
Installed True
PNP Device ID
PCI\VEN_1022&DEV_2000&SUBSYS_20001014&REV_44\3&267A616A&0&10
Last Reset 7/20/2001 6:30:05 AM
Index 0
Service Name PCNet5
IP Address 192.168.91.1
IP Subnet 255.255.255.0
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:02:55:58:14:95
Service Name PCnet
IRQ Number 27
I/O Port 0x2000-0x201F
Driver c:\winnt\system32\drivers\pcntn5m.sys (29968, 4.09.00)

Name [00000004] RAS Async Adapter
Adapter Type Not Available
Product Name RAS Async Adapter
Installed True

PNP Device ID Not Available
Last Reset 7/20/2001 6:30:05 AM
Index 4
Service Name AsyncMac
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Not Available

Name [00000005] WAN Miniport (L2TP)
Adapter Type Not Available
Product Name WAN Miniport (L2TP)
Installed True
PNP Device ID ROOT\MS_L2TPMINIPORT\0000
Last Reset 7/20/2001 6:30:05 AM
Index 5
Service Name Rasl2tp
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Rasl2tp
Driver c:\winnt\system32\drivers\rasl2tp.sys (50800, 5.00.2179.1)

Name [00000006] WAN Miniport (PPTP)
Adapter Type Wide Area Network (WAN)
Product Name WAN Miniport (PPTP)
Installed True
PNP Device ID ROOT\MS_PPTPMINIPORT\0000
Last Reset 7/20/2001 6:30:05 AM
Index 6
Service Name PptpMiniport
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 50:50:54:50:30:30
Service Name PptpMiniport
Driver c:\winnt\system32\drivers\raspptp.sys (47856, 5.00.2160.1)

Name [00000007] Direct Parallel
Adapter Type Not Available
Product Name Direct Parallel
Installed True
PNP Device ID ROOT\MS_PTMINIPORT\0000
Last Reset 7/20/2001 6:30:05 AM
Index 7

Service Name Raspti
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name Raspti
Driver c:\winnt\system32\drivers\raspti.sys (16880, 5.00.2146.1)

Name [00000008] WAN Miniport (IP)
Adapter Type Not Available
Product Name WAN Miniport (IP)
Installed True
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 7/20/2001 6:30:05 AM
Index 8
Service Name NdisWan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name NdisWan
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000009] WAN Miniport (NetBEUI, Dial In)
Adapter Type Not Available
Product Name WAN Miniport (NetBEUI, Dial In)
Installed True
PNP Device ID ROOT\MS_NDISWANNBFIN\0000
Last Reset 7/20/2001 6:30:05 AM
Index 9
Service Name NdisWan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name NdisWan
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000010] WAN Miniport (NetBEUI, Dial In)
Adapter Type Not Available
Product Name WAN Miniport (NetBEUI, Dial In)
Installed True
PNP Device ID ROOT\MS_NDISWANNBFIN\0001
Last Reset 7/20/2001 6:30:05 AM
Index 10
Service Name NdisWan
IP Address Not Available

IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name NdisWan
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000011] WAN Miniport (NetBEUI, Dial Out)
Adapter Type Not Available
Product Name WAN Miniport (NetBEUI, Dial Out)
Installed True
PNP Device ID ROOT\MS_NDISWANNBFOUT\0000
Last Reset 7/20/2001 6:30:05 AM
Index 11
Service Name NdisWan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name NdisWan
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000012] WAN Miniport (NetBEUI, Dial Out)
Adapter Type Not Available
Product Name WAN Miniport (NetBEUI, Dial Out)
Installed True
PNP Device ID ROOT\MS_NDISWANNBFOUT\0001
Last Reset 7/20/2001 6:30:05 AM
Index 12
Service Name NdisWan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name NdisWan
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000013] WAN Miniport (NetBEUI, Dial Out)
Adapter Type Not Available
Product Name WAN Miniport (NetBEUI, Dial Out)
Installed True
PNP Device ID ROOT\MS_NDISWANNBFOUT\0002
Last Reset 7/20/2001 6:30:05 AM
Index 13
Service Name NdisWan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available

DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Service Name NdisWan
Driver c:\winnt\system32\drivers\ndiswan.sys (90768, 5.00.2184.1)

Name [00000014] cLAN Host Adapter
Adapter Type Ethernet 802.3
Product Name cLAN Host Adapter
Installed True
PNP Device ID
PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&13C0B0C5&0&28
Last Reset 7/20/2001 6:30:05 AM
Index 14
Service Name GNINDIS
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled True
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:90:FA:00:1C:8A
Service Name GNINDIS
IRQ Number 20
Driver c:\winnt\system32\drivers\gnindis.sys (22726, 4.2.0.75)

Name [00000015] Intel(R) PRO/100+ Server Adapter (PILA8470B)
Adapter Type Ethernet 802.3
Product Name Intel(R) PRO/100+ Server Adapter (PILA8470B)
Installed True
PNP Device ID
PCI\VEN_8086&DEV_1229&SUBSYS_100C8086&REV_08\3&267A616A&0&50
Last Reset 7/20/2001 6:30:05 AM
Index 15
Service Name E100B
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled False
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:D0:B7:BF:FE:BE
Service Name E100B
IRQ Number 18
I/O Port 0x2040-0x207F
Driver c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

[Protocol]

Item Value
Name MSAFD Tcpip [TCP/IP]
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True

MaximumAddressSize 16 bytes
MaximumMessageSize 0 bytes
MessageOriented False
MinimumAddressSize 16 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData True
SupportsGracefulClosing True
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD Tcpip [UDP/IP]
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 16 bytes
MaximumMessageSize 65467 bytes
MessageOriented True
MinimumAddressSize 16 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting True

Name RSVP UDP Service Provider
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 16 bytes
MaximumMessageSize 65467 bytes
MessageOriented True
MinimumAddressSize 16 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption True
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting True

Name RSVP TCP Service Provider
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 16 bytes
MaximumMessageSize 0 bytes
MessageOriented False
MinimumAddressSize 16 bytes
PseudoStreamOriented False

SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption True
SupportsExpeditedData True
SupportsGracefulClosing True
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_{BAC8900B-457B-4294-A0ED-FD124E8A1212}]
SEQPACKET 17
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_{BAC8900B-457B-4294-A0ED-FD124E8A1212}]
DATAGRAM 17
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_{7488D6FD-F754-46DC-9D28-8B4824EE0DC5}]
SEQPACKET 15
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False

SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_{7488D6FD-F754-46DC-9D28-8B4824EE0DC5}]
DATAGRAM 15
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_{FCA901D1-07B8-4C5C-A941-E0F04B1E76EB}]
SEQPACKET 6
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_{FCA901D1-07B8-4C5C-A941-E0F04B1E76EB}]
DATAGRAM 6
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False

SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfIn{82E99CDB-174B-4E11-BA71-18DCCE0A24AE}] SEQPACKET 10
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfIn{82E99CDB-174B-4E11-BA71-18DCCE0A24AE}] DATAGRAM 10
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfIn{CDF4FCFD-9D61-47C6-9F00-26205FD125B3}] SEQPACKET 11
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False

SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfIn{CDF4FCFD-9D61-47C6-9F00-26205FD125B3}] DATAGRAM 11
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfOut{41802B0C-7E2A-494F-9B6C-3FB781035176}] SEQPACKET 12
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfOut{41802B0C-7E2A-494F-9B6C-3FB781035176}] DATAGRAM 12
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

```
Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfOut{71CB1DF0-3F1B-457A-B7C5-4F5FDE9C8A5F}] SEQUENCE 13
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False
```

```
Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfOut{71CB1DF0-3F1B-457A-B7C5-4F5FDE9C8A5F}] DATAGRAM 13
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False
```

```
Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfOut{B5D0BAD9-B12C-43B9-B40E-D89AA5241F84}] SEQUENCE 14
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False
```

```
Name MSAFD NetBIOS [\Device\Nbf_NdisWanNbfOut{B5D0BAD9-B12C-43B9-B40E-D89AA5241F84}] DATAGRAM 14
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False
```

```
Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{BAC8900B-457B-4294-A0ED-FD124E8A1212}] SEQUENCE 18
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False
```

```
Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{BAC8900B-457B-4294-A0ED-FD124E8A1212}] DATAGRAM 18
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False
```

```
Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{7488D6FD-F754-46DC-9D28-8B4824EE0DC5}] SEQUENCE 16
```

ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{7488D6FD-F754-46DC-9D28-8B4824EE0DC5}] DATAGRAM 16
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{FCA901D1-07B8-4C5C-A941-E0F04B1E76EB}] SEQPACKET 0
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{FCA901D1-07B8-4C5C-A941-E0F04B1E76EB}] DATAGRAM 0
ConnectionlessService True
GuaranteesDelivery False

GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{9CC59D60-6563-4459-B51A-CFE8755EAA2C}] SEQPACKET 4
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{9CC59D60-6563-4459-B51A-CFE8755EAA2C}] DATAGRAM 4
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C0FE4CDF-79B0-4356-B3E7-BF4B29CC58E9}] SEQPACKET 5
ConnectionlessService False
GuaranteesDelivery True
GuaranteesSequencing True
MaximumAddressSize 20 bytes

```

MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting False
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

```

```

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{C0FE4CDF-79B0-4356-B3E7-
BF4B29CC58E9}] DATAGRAM 5
ConnectionlessService True
GuaranteesDelivery False
GuaranteesSequencing False
MaximumAddressSize 20 bytes
MaximumMessageSize 64000 bytes
MessageOriented True
MinimumAddressSize 20 bytes
PseudoStreamOriented False
SupportsBroadcasting True
SupportsConnectData False
SupportsDisconnectData False
SupportsEncryption False
SupportsExpeditedData False
SupportsGracefulClosing False
SupportsGuaranteedBandwidth False
SupportsMulticasting False

```

[WinSock]

```

Item Value
File c:\winnt\system32\winsock.dll
Version 3.10
Size 2.80 KB (2,864 bytes)

```

```

File c:\winnt\system32\wsock32.dll
Version 5.00.2152.1
Size 21.27 KB (21,776 bytes)

```

[Ports]

[Following are sub-categories of this main category]

[Serial]

```

Item Value
Name COM1
Status OK
PNP Device ID ACPI\PNP0501\1
Maximum Input Buffer Size Not Available
Maximum Output Buffer Size Not Available
Settable Baud Rate Not Available
Settable Data Bits Not Available

```

```

Settable Flow Control Not Available
Settable Parity Not Available
Settable Parity Check Not Available
Settable Stop Bits Not Available
Settable RLSD Not Available
Supports RLSD Not Available
Supports 16 Bit Mode Not Available
Supports Special Characters Not Available
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy -1
Abort Read/Write on Error Not Available
Binary Mode Enabled Not Available
Continue Xmit on XOff Not Available
CTS Outflow Control Not Available
Discard NULL Bytes Not Available
DSR Outflow Control Not Available
DSR Sensitivity Not Available
DTR Flow Control Type Not Available
EOF Character Not Available
Error Replace Character Not Available
Error Replacement Enabled Not Available
Event Character Not Available
Parity Check Enabled -1
RTS Flow Control Type Not Available
XOff Character 19
XOffXmit Threshold 512
XOn Character 17
XOnXmit Threshold 2048
XOnXOff InFlow Control Not Available
XOnXOff OutFlow Control Not Available
IRQ Number 4
I/O Port 0x03F8-0x03FF
Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

```

```

Name COM2
Status OK
PNP Device ID ACPI\PNP0501\2
Maximum Input Buffer Size 0
Maximum Output Buffer Size False
Settable Baud Rate True
Settable Data Bits True
Settable Flow Control True
Settable Parity True
Settable Parity Check True
Settable Stop Bits True
Settable RLSD True
Supports RLSD True
Supports 16 Bit Mode False
Supports Special Characters False
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy 0
Abort Read/Write on Error 0
Binary Mode Enabled -1

```

```

Continue XMit on XOff 0
CTS Outflow Control 0
Discard NULL Bytes 0
DSR Outflow Control 0
DSR Sensitivity 0
DTR Flow Control Type Enable
EOF Character 0
Error Replace Character 0
Error Replacement Enabled 0
Event Character 0
Parity Check Enabled 0
RTS Flow Control Type Enable
XOff Character 19
XOffXMit Threshold 512
XOn Character 17
XOnXMit Threshold 2048
XOnXOff InFlow Control 0
XOnXOff OutFlow Control 0
IRQ Number 3
I/O Port 0x02F8-0x02FF
Driver c:\winnt\system32\drivers\serial.sys (62448, 5.00.2134.1)

```

[Parallel]

```

Item Value
Name LPT1
PNP Device ID ACPI\PNP0400\1

```

[Storage]

[Following are sub-categories of this main category]

[Drives]

```

Item Value
Drive A:
Description 3 1/2 Inch Floppy Drive

Drive C:
Description Local Fixed Disk
Compressed False
File System NTFS
Size 16.94 GB (18,194,284,544 bytes)
Free Space 15.19 GB (16,309,764,096 bytes)
Volume Name
Volume Serial Number E0561636
Partition Disk #0, Partition #0
Partition Size 16.94 GB (18,194,287,104 bytes)
Starting Offset 32256 bytes
Drive Description Disk drive
Drive Manufacturer (Standard disk drives)
Drive Model IBM-PSG ST318404LC !# SCSI Disk Device
Drive BytesPerSector 512
Drive MediaLoaded True
Drive MediaType Fixed hard disk media
Drive Partitions 1
Drive SCSI Bus 0

```

```

Drive SCSILogicalUnit 0
Drive SCSIPort 2
Drive SCSTargetId 0
Drive SectorsPerTrack 63
Drive Size 18194319360 bytes
Drive TotalCylinders 2212
Drive TotalSectors 35535780
Drive TotalTracks 564060
Drive TracksPerCylinder 255

```

[SCSI]

```

Item Value
Name Adaptec AIC-7899 Ultra160/m PCI SCSI Card
Caption Adaptec AIC-7899 Ultra160/m PCI SCSI Card
Driver adpul60m
Status OK
PNP Device ID
PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13C0B0C5&0&18
Device ID
PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13C0B0C5&0&18
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 28
I/O Port 0x2100-0x21FF
Driver c:\winnt\system32\drivers\adpul60m.sys (64432, v3.10a)

```

```

Name Adaptec AIC-7899 Ultra160/m PCI SCSI Card
Caption Adaptec AIC-7899 Ultra160/m PCI SCSI Card
Driver adpul60m
Status OK
PNP Device ID
PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13C0B0C5&0&19
Device ID
PCI\VEN_9005&DEV_00CF&SUBSYS_019D1014&REV_01\3&13C0B0C5&0&19
Device Map Not Available
Index Not Available
Max Number Controlled Not Available
IRQ Number 29
I/O Port 0x2200-0x22FF
Driver c:\winnt\system32\drivers\adpul60m.sys (64432, v3.10a)

```

[Printing]

```

Name Port Name Server Name
No printing information

```

[Problem Devices]

```

Device PNP Device ID Error Code
Not Available ACPI\IBM37C0\4&F0B8F99&0 28

```

[USB]

```

Device PNP Device ID

```

Standard OpenHCD USB Host Controller
 PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_04\3&267A616A0&7A
 USB Root Hub USB\ROOT_HUB\4&372644EA&0

[Software Environment]

[Following are sub-categories of this main category]

[Drivers]

Name	Description	File	Type	Started	Start	Mode	State	Status
	Error Control	Accept	Pause	Accept	Stop			
abiosdsk	Abiosdsk	Not Available	Kernel Driver	False	False	False	False	False
	Disabled	Stopped OK	Ignore	False	False	False	False	False
abp480n5	abp480n5	Not Available	Kernel Driver	False	False	False	False	False
	Disabled	Stopped OK	Normal	False	False	False	False	False
acpi	Microsoft ACPI Driver	c:\winnt\system32\drivers\acpi.sys	Kernel Driver	True	Running OK	Normal	False	True
acpiec	ACPIEC	c:\winnt\system32\drivers\acpiec.sys	Kernel Driver	False	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	False	False	False
adpu160m	adpu160m	c:\winnt\system32\drivers\adpu160m.sys	Kernel Driver	True	Running OK	Normal	False	True
afd	AFD Networking Support	c:\winnt\system32\drivers\afd.sys	Kernel Driver	True	Running OK	Normal	False	True
	Running OK	Normal	False	True	False	Disabled		
aha154x	Aha154x	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
aic116x	aic116x	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
aic78u2	aic78u2	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
aic78xx	aic78xx	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
ami0nt	ami0nt	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
amsint	amsint	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
asc	asc	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
asc3350p	asc3350p	Not Available	Kernel Driver	False	Disabled	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled		
asc3550	asc3550	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Normal	False	False	False	Disabled		
asynctac	RAS Asynchronous Media Driver	c:\winnt\system32\drivers\asynctac.sys	Kernel Driver	False	Manual	Stopped OK	Normal	False
	Manual	Stopped OK	Normal	False	False	Disabled		
atapi	Standard IDE/ESDI Hard Disk Controller	c:\winnt\system32\drivers\atapi.sys	Kernel Driver	True	Running OK	Normal	False	True
	Running OK	Normal	False	True	False	Disabled		
atdisk	Atdisk	Not Available	Kernel Driver	False	Stopped OK	Normal	False	False
	Stopped OK	Ignore	False	False	False	Disabled		
atmarpc	ATM ARP Client Protocol	c:\winnt\system32\drivers\atmarpc.sys	Kernel Driver	False	Manual	Stopped OK	Normal	False
	Manual	Stopped OK	Normal	False	False	Disabled		
audstub	Audio Stub Driver	c:\winnt\system32\drivers\audstub.sys	Kernel Driver	True	Running OK	Normal	False	True
	Running OK	Normal	False	True	False	Disabled		
beep	Beep	c:\winnt\system32\drivers\beep.sys	Kernel Driver	True	System	Running OK	Normal	False
	System	Running OK	Normal	False	True	Disabled		

buslogic	BusLogic	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
cd20xrnt	cd20xrnt	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
cdaudio	Cdaudio	c:\winnt\system32\drivers\cdaudio.sys	Kernel Driver	False	System	Stopped OK	Ignore	False	False
	System	Stopped OK	Ignore	False	False	Disabled			
cdfs	Cdfs	c:\winnt\system32\drivers\cdfs.sys	File System Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
cdrom	CD-ROM Driver	c:\winnt\system32\drivers\cdrom.sys	Kernel Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
changer	Changer	Not Available	Kernel Driver	False	Ignore	False	False	System	Stopped OK
	Ignore	False	False	System	Stopped OK	Disabled			
cpqarray	Cpqarray	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
cpqarray2	cpqarray2	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
cpqfcalm	cpqfcalm	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
cpqfws2e	cpqfws2e	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
dac960nt	dac960nt	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
deckzpsx	deckzpsx	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
dfsdriver	DfsDriver	c:\winnt\system32\drivers\dfs.sys	File System Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
disk	Disk Driver	c:\winnt\system32\drivers\disk.sys	Kernel Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
diskperf	Diskperf	c:\winnt\system32\drivers\diskperf.sys	Kernel Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
dmboot	dmboot	c:\winnt\system32\drivers\dmboot.sys	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
dmio	Logical Disk Manager Driver	c:\winnt\system32\drivers\dmio.sys	Kernel Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
dmload	dmload	c:\winnt\system32\drivers\dmload.sys	Kernel Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
e1000	Intel(R) PRO/1000 Gigabit Server Adapter Driver	c:\winnt\system32\drivers\e1000nt5.sys	Kernel Driver	False	Manual	Stopped OK	Normal	False	False
	Manual	Stopped OK	Normal	False	False	Disabled			
e100b	Intel PRO Adapter Driver	c:\winnt\system32\drivers\e100bnt5.sys	Kernel Driver	True	Manual	Running OK	Normal	False	True
	Manual	Running OK	Normal	False	True	Disabled			
efs	EFS	c:\winnt\system32\drivers\efs.sys	File System Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
fastfat	Fastfat	c:\winnt\system32\drivers\fastfat.sys	File System Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
fd16_700	Fd16_700	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
fdc	Floppy Disk Controller Driver	c:\winnt\system32\drivers\fdc.sys	Kernel Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			
fireport	fireport	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
flashpnt	flashpnt	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False
	Disabled	Stopped OK	Normal	False	False	Disabled			
flpydisk	Floppy Disk Driver	c:\winnt\system32\drivers\flpydisk.sys	Kernel Driver	True	Manual	Running OK	Normal	False	True
	Manual	Running OK	Normal	False	True	Disabled			
ftdisk	Volume Manager Driver	c:\winnt\system32\drivers\ftdisk.sys	Kernel Driver	True	Running OK	Normal	False	True	
	True	Running OK	Normal	False	True	Disabled			

gnindis	cLAN NDIS Driver	c:\winnt\system32\drivers\gnindis.sys	Kernel Driver	True	Auto	Running	OK	Normal	False	True
gnivia	cLAN VIA Driver	c:\winnt\system32\drivers\gnivia.sys	Kernel Driver	True	Auto	Running	OK	Normal	False	True
gpc	Generic Packet Classifier	c:\winnt\system32\drivers\msgpc.sys	Kernel Driver	True	Manual	Running	OK	Normal	False	True
hptx	HP 10/100TX PCI LAN Adapter NT Driver	c:\winnt\system32\drivers\hptxnt5.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver	c:\winnt\system32\drivers\i8042prt.sys	Kernel Driver	True	System	Running	OK	Normal	False	True
ini910u	ini910u Not Available	Kernel Driver	False	Disabled	Stopped	OK	Normal	False	False	False
intelide	IntelIde	Not Available	Kernel Driver	False	Disabled	Stopped	OK	Normal	False	False
ipfilterdriver	IP Traffic Filter Driver	c:\winnt\system32\drivers\ipfltdrv.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
ipinip	IP in IP Tunnel Driver	c:\winnt\system32\drivers\ipinip.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
ipnat	IP Network Address Translator	c:\winnt\system32\drivers\ipnat.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
ipsec	IPSEC driver	c:\winnt\system32\drivers\ipsec.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
ipsraidn	ipsraidn	Not Available	Kernel Driver	False	Disabled	Stopped	OK	Normal	False	False
isapnp	PnP ISA/EISA Bus Driver	c:\winnt\system32\drivers\isapnp.sys	Kernel Driver	True	Boot	Running	OK	Critical	False	True
kbdclass	Keyboard Class Driver	c:\winnt\system32\drivers\kbdclass.sys	Kernel Driver	True	System	Running	OK	Normal	False	True
ksecdd	KSecDD	c:\winnt\system32\drivers\ksecdd.sys	Kernel Driver	True	Boot	Running	OK	Normal	False	True
lbrtfdc	lbrtfdc	Not Available	Kernel Driver	False	System	Stopped	OK	Ignore	False	False
lp6nds35	lp6nds35	Not Available	Kernel Driver	False	Disabled	Stopped	OK	Normal	False	False
mmdd	mmdd	c:\winnt\system32\drivers\mmdd.sys	Kernel Driver	True	System	Running	OK	Ignore	False	True
modem	Modem	c:\winnt\system32\drivers\modem.sys	Kernel Driver	False	Manual	Stopped	OK	Ignore	False	False
mouclass	Mouse Class Driver	c:\winnt\system32\drivers\mouclass.sys	Kernel Driver	True	System	Running	OK	Normal	False	True
mountmgr	MountMgr	c:\winnt\system32\drivers\mountmgr.sys	Kernel Driver	True	Boot	Running	OK	Normal	False	True
mrraid35x	mrraid35x	Not Available	Kernel Driver	False	Disabled	Stopped	OK	Normal	False	False
mrxsmb	MRXSMB	c:\winnt\system32\drivers\mrxsmb.sys	File System Driver	True	System	Running	OK	Normal	False	True
msfs	Msfs	c:\winnt\system32\drivers\msfs.sys	File System Driver	True	System	Running	OK	Normal	False	True
mskssrv	Microsoft Streaming Service Proxy	c:\winnt\system32\drivers\mskssrv.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
mspclock	Microsoft Streaming Clock Proxy	c:\winnt\system32\drivers\mspclock.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
mspqm	Microsoft Streaming Quality Manager Proxy	c:\winnt\system32\drivers\mspqm.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
mup	Mup	c:\winnt\system32\drivers\mup.sys	File System Driver	True	Boot	Running	OK	Normal	False	True
nbf	NetBEUI Protocol	c:\winnt\system32\drivers\nbf.sys	Kernel Driver	True	Auto	Running	OK	Normal	False	True
ncrc710	Ncrc710 Not Available	Kernel Driver	False	Disabled	Stopped	OK	Normal	False	False	False
ndis	NDIS System Driver	c:\winnt\system32\drivers\ndis.sys	Kernel Driver	True	Boot	Running	OK	Normal	False	True
ndistapi	Remote Access NDIS TAPI Driver	c:\winnt\system32\drivers\ndistapi.sys	Kernel Driver	True	Manual	Running	OK	Normal	False	True
ndiswan	Remote Access NDIS WAN Driver	c:\winnt\system32\drivers\ndiswan.sys	Kernel Driver	True	Manual	Running	OK	Normal	False	True
ndproxy	NDIS Proxy	c:\winnt\system32\drivers\ndproxy.sys	Kernel Driver	True	Manual	Running	OK	Normal	False	True
netbios	NetBIOS Interface	c:\winnt\system32\drivers\netbios.sys	File System Driver	True	System	Running	OK	Normal	False	True
netbt	NetBios over Tcpip	c:\winnt\system32\drivers\netbt.sys	Kernel Driver	True	System	Running	OK	Normal	False	True
netdetect	NetDetect	c:\winnt\system32\drivers\netdetect.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
npfs	Npfs	c:\winnt\system32\drivers\npfs.sys	File System Driver	True	System	Running	OK	Normal	False	True
ntfs	Ntfs	c:\winnt\system32\drivers\ntfs.sys	File System Driver	True	True	Disabled	Running	OK	Normal	False
null	Null	c:\winnt\system32\drivers\null.sys	Kernel Driver	True	System	Running	OK	Normal	False	True
nwlkflt	IPX Traffic Filter Driver	c:\winnt\system32\drivers\nwlkflt.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
nwlkfld	IPX Traffic Forwarder Driver	c:\winnt\system32\drivers\nwlkfld.sys	Kernel Driver	False	Manual	Stopped	OK	Normal	False	False
openhci	Microsoft USB Open Host Controller Driver	c:\winnt\system32\drivers\openhci.sys	Kernel Driver	True	Manual	Running	OK	Normal	False	True
parallel	Parallel class driver	c:\winnt\system32\drivers\parallel.sys	Kernel Driver	True	Manual	Running	OK	Normal	False	True
parport	Parallel port driver	c:\winnt\system32\drivers\parport.sys	Kernel Driver	True	System	Running	OK	Ignore	False	True
partmgr	PartMgr	c:\winnt\system32\drivers\partmgr.sys	Kernel Driver	True	Boot	Running	OK	Normal	False	True
parvdm	ParVdm	c:\winnt\system32\drivers\parvdm.sys	Kernel Driver	True	Auto	Running	OK	Ignore	False	True
pci	PCI Bus Driver	c:\winnt\system32\drivers\pci.sys	Kernel Driver	True	Boot	Running	OK	Critical	False	True
pcidump	PCIDump	Not Available	Kernel Driver	False	System	Stopped	OK	Ignore	False	False
pciide	PCIIde	c:\winnt\system32\drivers\pciide.sys	Kernel Driver	True	Boot	Running	OK	Normal	False	True

pcmcia	Pcmcia	c:\winnt\system32\drivers\pcmcia.sys	Kernel Driver	False	Stopped OK	Normal	False	False	spud	Special Purpose Utility Driver	c:\winnt\system32\drivers\spud.sys	Kernel Driver	False	Manual
pcnet	AMD PCNET Compatable Adapter Driver	c:\winnt\system32\drivers\pcntn5m.sys	Kernel Driver	True	Running OK	Normal	False	True	Stopped OK	Normal	False	False	False	False
pdcomp	PDCOMP	Not Available	Kernel Driver	False	Manual	Stopped OK	OK	swenum	Software Bus Driver	c:\winnt\system32\drivers\swenum.sys	Kernel Driver	True	Manual	Running OK
pdframe	PDFFRAME	Not Available	Kernel Driver	False	Manual	Stopped OK	OK	symc810	symc810	Not Available	Kernel Driver	False	Disabled	True
pdreli	PDRELI	Not Available	Kernel Driver	False	Manual	Stopped OK	OK	symc8xx	symc8xx	Not Available	Kernel Driver	False	Disabled	True
pdrframe	PDRFRAME	Not Available	Kernel Driver	False	Manual	Stopped OK	OK	sym_hi	sym_hi	Not Available	Kernel Driver	False	Disabled	True
pptpminiport	WAN Miniport (PPTP)	c:\winnt\system32\drivers\raspttp.sys	Kernel Driver	True	Running OK	Normal	False	True	tcpip	TCP/IP Protocol Driver	c:\winnt\system32\drivers\tdcpip.sys	Kernel Driver	True	System
ptilink	Direct Parallel Link Driver	c:\winnt\system32\drivers\ptilink.sys	Kernel Driver	True	Running OK	Normal	False	True	tdasync	TDASync	c:\winnt\system32\drivers\tdasync.sys	Kernel Driver	False	Manual
ql1080	ql1080	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	tdipx	TDIPX	c:\winnt\system32\drivers\tdipx.sys	Kernel Driver	False	Manual
ql10wnt	Ql10wnt	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	tdnetb	TDNETB	c:\winnt\system32\drivers\tdnetb.sys	Kernel Driver	False	Manual
ql1240	ql1240	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	tdpipe	TDPIPE	c:\winnt\system32\drivers\tdpipe.sys	Kernel Driver	False	Manual
ql2100	ql2100	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	tdspx	TDSPX	c:\winnt\system32\drivers\tdspx.sys	Kernel Driver	False	Manual
rasacd	Remote Access Auto Connection Driver	c:\winnt\system32\drivers\rasacd.sys	Kernel Driver	True	Running OK	Normal	False	True	tdtcp	TDTCP	c:\winnt\system32\drivers\tdtcp.sys	Kernel Driver	False	Manual
rasl2tp	WAN Miniport (L2TP)	c:\winnt\system32\drivers\rasl2tp.sys	Kernel Driver	True	Running OK	Normal	False	True	termdd	Terminal Device Driver	c:\winnt\system32\drivers\termdd.sys	Kernel Driver	True	Auto
raspti	Direct Parallel	c:\winnt\system32\drivers\raspti.sys	Kernel Driver	True	Running OK	Normal	False	True	tga	tga	Not Available	Kernel Driver	False	System
rca	Microsoft Streaming Network Raw Channel Access	c:\winnt\system32\drivers\rca.sys	Kernel Driver	False	Manual	Stopped OK	Normal	False	udfs	Udfs	c:\winnt\system32\drivers\udfs.sys	File System Driver	False	Disabled
rdbss	Rdbss	c:\winnt\system32\drivers\rdbss.sys	File System Driver	True	System	Running OK	Normal	False	True	ultra66	ultra66	Not Available	Kernel Driver	False
rdpwd	RDPWD	c:\winnt\system32\drivers\rdpwd.sys	Kernel Driver	False	Manual	Stopped OK	Normal	False	True	update	Microcode Update Driver	c:\winnt\system32\drivers\update.sys	Kernel Driver	True
redbook	Digital CD Audio Playback Filter Driver	c:\winnt\system32\drivers\redbook.sys	Kernel Driver	False	System	Stopped OK	Normal	False	True	usbhub	Microsoft USB Standard Hub Driver	c:\winnt\system32\drivers\usbhub.sys	Kernel Driver	True
s3inc	S3Inc	c:\winnt\system32\drivers\s3sav4m.sys	Kernel Driver	True	Manual	Running OK	Ignore	False	True	vgasave	VgaSave	c:\winnt\system32\drivers\vga.sys	Kernel Driver	True
serenum	Serenum Filter Driver	c:\winnt\system32\drivers\serenum.sys	Kernel Driver	True	Manual	Running OK	Normal	False	True	wanarp	Remote Access IP ARP Driver	c:\winnt\system32\drivers\wanarp.sys	Kernel Driver	True
serial	Serial port driver	c:\winnt\system32\drivers\serial.sys	Kernel Driver	True	System	Running OK	Ignore	False	True	wdica	WDICA	Not Available	Kernel Driver	False
sfloppy	Sfloppy	c:\winnt\system32\drivers\sfloppy.sys	Kernel Driver	False	System	Stopped OK	Ignore	False	False	[Environment Variables]				
sglfb	sglfb	Not Available	Kernel Driver	False	System	Stopped OK	Normal	False	False	Variable	Value	User Name		
simbad	Simbad	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False	ComSpec	%SystemRoot%\system32\cmd.exe	<SYSTEM>		
sparrow	Sparrow	Not Available	Kernel Driver	False	Disabled	Stopped OK	Normal	False	False	NUMBER_OF_PROCESSORS	2	<SYSTEM>		
										OS	Windows_NT	<SYSTEM>		
										Os2LibPath	%SystemRoot%\system32\os2\dll;	<SYSTEM>		
										Path	%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;C:\Program Files\Microsoft SQL Server\80\Tools\BINN	<SYSTEM>		
										PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH	<SYSTEM>		
										PROCESSOR_ARCHITECTURE	x86	<SYSTEM>		

```

PROCESSOR_IDENTIFIER x86 Family 6 Model 8 Stepping 6, GenuineIntel
<SYSTEM>
PROCESSOR_LEVEL 6 <SYSTEM>
PROCESSOR_REVISION 0806 <SYSTEM>
TEMP %SystemRoot%\TEMP <SYSTEM>
TMP %SystemRoot%\TEMP <SYSTEM>
WEBSPPNUMBER 1 <SYSTEM>
windir %SystemRoot% <SYSTEM>
TEMP %USERPROFILE%\Local Settings\Temp CLIENT1\Administrator
TMP %USERPROFILE%\Local Settings\Temp CLIENT1\Administrator

```

[Jobs]

[Following are sub-categories of this main category]

[Print]

Document	Size	Owner	Notify	Status	Time Submitted	Start Time	Until Time	Elapsed Time	Pages Printed	Job ID	Priority	Parameters	Driver Name	Print Processor	Host Print
Queue	Data Type	Name													
Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown	Unknown

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
No network connections information				

[Running Tasks]

Name	Path	Process ID	Priority	Min Working Set	Max Working Set
Working Set	Start Time	Version	Size	File Date	
system idle process		Not Available	0	0	Not Available
Available	Not Available	Unknown	Unknown	Unknown	
system	Not Available	8	0	1413120	Not Available
	Unknown	Unknown	Unknown		
smss.exe	c:\winnt\system32\smss.exe	164	11	204800	
bytes)	1413120 7/20/2001 1:30:32 PM	5.00.2170.1	44.27 KB	(45,328	
	12/7/1999 4:00:00 AM				
csrss.exe	c:\winnt\system32\csrss.exe	188	13	204800	
bytes)	1413120 7/20/2001 1:30:34 PM	5.00.2137.1	5.27 KB	(5,392 bytes)	
	12/7/1999 4:00:00 AM				
winlogon.exe	c:\winnt\system32\winlogon.exe	184	13	204800	
bytes)	1413120 7/20/2001 1:30:35 PM	5.00.2182.1	173.27 KB	(177,424	
	12/7/1999 4:00:00 AM				
services.exe	c:\winnt\system32\services.exe	240	9	204800	
bytes)	1413120 7/20/2001 1:30:36 PM	5.00.2134.1	86.77 KB	(88,848	
	12/7/1999 4:00:00 AM				
lsass.exe	c:\winnt\system32\lsass.exe	252	13	204800	
bytes)	1413120 7/20/2001 1:30:36 PM	5.00.2184.1	32.77 KB	(33,552	
	12/7/1999 4:00:00 AM				
gnconmgr.exe	c:\winnt\system32\gnconmgr.exe	368	8	204800	
bytes)	1413120 7/20/2001 1:30:38 PM	4.2.0.75	168.05 KB	(172,079	
	6/5/2001 1:12:35 PM				
svchost.exe	c:\winnt\system32\svchost.exe	420	8	204800	
bytes)	1413120 7/20/2001 1:30:38 PM	5.00.2134.1	7.77 KB	(7,952 bytes)	
	12/7/1999 4:00:00 AM				

```

msdtc.exe c:\winnt\system32\msdtc.exe 452 8 204800
1413120 7/20/2001 1:30:42 PM 1999.9.3421.3 6.77 KB (6,928 bytes)
1/16/2001 7:07:02 AM
svchost.exe c:\winnt\system32\svchost.exe 588 8 204800
1413120 7/20/2001 1:30:47 PM 5.00.2134.1 7.77 KB (7,952 bytes)
12/7/1999 4:00:00 AM
regsvc.exe c:\winnt\system32\regsvc.exe 616 8 204800
1413120 7/20/2001 1:30:47 PM 5.00.2155.1 65.27 KB (66,832
bytes) 12/7/1999 4:00:00 AM
mstask.exe c:\winnt\system32\mstask.exe 632 8 204800
1413120 7/20/2001 1:30:47 PM 4.71.2137.1 115.27 KB (118,032
bytes) 1/16/2001 2:59:30 PM
winmgmt.exe c:\winnt\system32\wbem\winmgmt.exe 640 8 204800
1413120 7/20/2001 1:30:48 PM 1.50.1085.0001 188.05 KB (192,567
bytes) 12/7/1999 4:00:00 AM
inetinfo.exe c:\winnt\system32\inetinfo.exe 764 8
204800 1413120 7/20/2001 1:30:48 PM 5.00.0984 14.27 KB
(14,608 bytes) 1/16/2001 7:07:17 AM
explorer.exe c:\winnt\explorer.exe 940 8 204800 1413120
7/20/2001 1:31:08 PM 5.00.2920.0000 232.77 KB (238,352 bytes)
12/7/1999 4:00:00 AM
rhostsrvr.exe c:\rhost\rhostsrvr.exe 988 8 204800 1413120
7/20/2001 1:31:10 PM Not Available 54.50 KB (55,808 bytes)
1/16/2001 6:29:11 PM
rsrvp.exe c:\winnt\system32\rsrvp.exe 1036 8 204800
1413120 7/20/2001 2:25:00 PM 5.00.2167.1 172.77 KB (176,912
bytes) 12/7/1999 4:00:00 AM

```

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer	Path
traffic.dll	5.00.2139.1	30.77 KB	(31,504 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\traffic.dll			
rsrvp.exe	5.00.2167.1	172.77 KB	(176,912 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rsrvp.exe			
rhostsrvr.exe	Not Available	54.50 KB	(55,808 bytes)		1/16/2001
6:29:11 PM	Not Available	c:\rhost\rhostsrvr.exe			
mstask.dll	4.71.2137.1	213.27 KB	(218,384 bytes)		1/16/2001
2:59:30 PM	Microsoft Corporation	c:\winnt\system32\mstask.dll			
wininet.dll	5.00.2920.0000	456.77 KB	(467,728 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\wininet.dll			
comdlg32.dll	5.00.2920.0000	236.77 KB	(242,448 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\comdlg32.dll			
netcfgx.dll	5.00.2175.1	533.77 KB	(546,576 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\netcfgx.dll			
rasdlg.dll	5.00.2194.1	514.27 KB	(526,608 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rasdlg.dll			
urlmon.dll	5.00.2920.0000	426.77 KB	(437,008 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\urlmon.dll			
ntshrui.dll	5.00.2134.1	46.77 KB	(47,888 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntshrui.dll			
linkinfo.dll	5.00.2134.1	15.77 KB	(16,144 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\linkinfo.dll			
msi.dll	1.10.1029.0	1.71 MB	(1,794,320 bytes)		12/7/1999 4:00:00 AM
	Microsoft Corporation	c:\winnt\system32\msi.dll			
powrprof.dll	5.00.2920.0000	13.27 KB	(13,584 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\powrprof.dll			
batmeter.dll	5.00.2920.0000	20.27 KB	(20,752 bytes)		12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\batmeter.dll			

```

stobject.dll 5.00.2144.1 81.77 KB (83,728 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\stobject.dll
webcheck.dll 5.00.2920.0000 251.77 KB (257,808 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\webcheck.dll
browselc.dll 5.00.2920.0000 34.50 KB (35,328 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\browselc.dll
netui1.dll 5.00.2134.1 210.27 KB (215,312 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\netui1.dll
netui0.dll 5.00.2134.1 70.27 KB (71,952 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\netui0.dll
ntlanman.dll 5.00.2157.1 35.27 KB (36,112 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\ntlanman.dll
browseui.dll 5.00.2920.0000 793.27 KB (812,304 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\browseui.dll
shdocvw.dll 5.00.2920.0000 1.05 MB (1,104,144 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\shdocvw.dll
explorer.exe 5.00.2920.0000 232.77 KB (238,352 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\explorer.exe
security.dll 5.00.2154.1 5.77 KB (5,904 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\security.dll
svcxext.dll 5.00.0984 39.77 KB (40,720 bytes) 1/16/2001
7:07:18 AM Microsoft Corporation c:\winnt\system32\inetsrv\svcxext.dll
admexs.dll 5.00.0984 27.77 KB (28,432 bytes) 1/16/2001
7:07:17 AM Microsoft Corporation c:\winnt\system32\inetsrv\admexs.dll
wamreg.dll 5.00.0984 46.27 KB (47,376 bytes) 1/16/2001
7:07:24 AM Microsoft Corporation c:\winnt\system32\inetsrv\wamreg.dll
metadata.dll 5.00.0984 70.77 KB (72,464 bytes) 1/16/2001
7:07:18 AM Microsoft Corporation
c:\winnt\system32\inetsrv\metadata.dll
iismap.dll 5.00.0984 56.27 KB (57,616 bytes) 1/16/2001
7:07:19 AM Microsoft Corporation c:\winnt\system32\iismap.dll
nsepm.dll 5.00.0984 43.27 KB (44,304 bytes) 1/16/2001
7:07:18 AM Microsoft Corporation c:\winnt\system32\inetsrv\nsepm.dll
admwprox.dll 5.00.0984 31.77 KB (32,528 bytes) 1/16/2001
7:07:19 AM Microsoft Corporation c:\winnt\system32\admwprox.dll
coadmin.dll 5.00.0984 39.77 KB (40,720 bytes) 1/16/2001
7:07:19 AM Microsoft Corporation
c:\winnt\system32\inetsrv\coadmin.dll
iisadmin.dll 5.00.0984 14.77 KB (15,120 bytes) 1/16/2001
7:07:17 AM Microsoft Corporation
c:\winnt\system32\inetsrv\iisadmin.dll
rpcref.dll 5.00.0984 4.27 KB (4,368 bytes) 1/16/2001 7:07:18 AM
Microsoft Corporation c:\winnt\system32\inetsrv\rpcref.dll
iisrtl.dll 5.00.0984 120.77 KB (123,664 bytes) 1/16/2001
7:07:18 AM Microsoft Corporation c:\winnt\system32\iisrtl.dll
inetinfo.exe 5.00.0984 14.27 KB (14,608 bytes) 1/16/2001
7:07:17 AM Microsoft Corporation
c:\winnt\system32\inetsrv\inetinfo.exe
rapilib.dll 5.00.2167.1 25.27 KB (25,872 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\rapilib.dll
rsvpsp.dll 5.00.2167.1 74.77 KB (76,560 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\rsvpsp.dll
ntmarta.dll 5.00.2158.1 98.77 KB (101,136 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\ntmarta.dll
provthrd.dll 1.50.1085.0000 68.07 KB (69,708 bytes) 1/16/2001
2:59:29 PM Microsoft Corporation c:\winnt\system32\wbem\provthrd.dll
ntevt.dll 1.50.1085.0000 192.06 KB (196,669 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\ntevt.dll

```

```

perfos.dll 5.00.2155.1 21.27 KB (21,776 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\perfos.dll
psapi.dll 5.00.2134.1 28.27 KB (28,944 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\psapi.dll
framedyn.dll 1.50.1085.0000 164.05 KB (167,992 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\framedyn.dll
cimwin32.dll 1.50.1085.0000 1.03 MB (1,077,306 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\cimwin32.dll
wbemsvc.dll 1.50.1085.0000 140.07 KB (143,430 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\wbemsvc.dll
wbemess.dll 1.50.1085.0001 352.05 KB (360,503 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\wbemess.dll
fastprox.dll 1.50.1085.0001 144.08 KB (147,534 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\fastprox.dll
wbemcore.dll 1.50.1085.0001 632.05 KB (647,224 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\wbemcore.dll
wbemcomn.dll 1.50.1085.0001 684.05 KB (700,472 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\wbemcomn.dll
winmgmt.exe 1.50.1085.0001 188.05 KB (192,567 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wbem\winmgmt.exe
msidle.dll 5.00.2920.0000 6.27 KB (6,416 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\msidle.dll
mstask.exe 4.71.2137.1 115.27 KB (118,032 bytes) 1/16/2001
2:59:30 PM Microsoft Corporation c:\winnt\system32\mstask.exe
regsvc.exe 5.00.2155.1 65.27 KB (66,832 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\regsvc.exe
wmi.dll 5.00.2191.1 6.27 KB (6,416 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\wmi.dll
netshell.dll 5.00.2176.1 456.77 KB (467,728 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\netshell.dll
netman.dll 5.00.2175.1 88.77 KB (90,896 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\netman.dll
sens.dll 5.00.2163.1 36.77 KB (37,648 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\sens.dll
es.dll 1999.9.3422.21 231.77 KB (237,328 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\es.dll
mtxoci.dll 1999.9.3421.3 109.27 KB (111,888 bytes) 1/16/2001
7:07:03 AM Microsoft Corporation c:\winnt\system32\mtxoci.dll
resutils.dll 5.00.2191.1 39.77 KB (40,720 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\resutils.dll
clusapi.dll 5.00.2179.1 50.27 KB (51,472 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\clusapi.dll
msvcp50.dll 5.00.7051 552.50 KB (565,760 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\msvcp50.dll
xolehlp.dll 1999.9.3421.3 17.27 KB (17,680 bytes) 1/16/2001
7:07:03 AM Microsoft Corporation c:\winnt\system32\xolehlp.dll
msdtclog.dll 1999.9.3421.3 89.77 KB (91,920 bytes) 1/16/2001
7:07:02 AM Microsoft Corporation c:\winnt\system32\msdtclog.dll
mtxclu.dll 1999.9.3421.3 50.27 KB (51,472 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\mtxclu.dll
msdtcprx.dll 1999.9.3422.10 619.27 KB (634,128 bytes) 1/16/2001
7:07:03 AM Microsoft Corporation c:\winnt\system32\msdtcprx.dll
txfaux.dll 1999.9.3422.24 341.27 KB (349,456 bytes) 1/16/2001
7:07:02 AM Microsoft Corporation c:\winnt\system32\txfaux.dll
msdtctm.dll 1999.9.3422.12 1.02 MB (1,070,864 bytes) 1/16/2001
7:07:03 AM Microsoft Corporation c:\winnt\system32\msdtctm.dll
msdtc.exe 1999.9.3421.3 6.77 KB (6,928 bytes) 1/16/2001 7:07:02 AM
Microsoft Corporation c:\winnt\system32\msdtc.exe

```

wshnetbs.dll	5.00.2134.1	7.77 KB (7,952 bytes)	12/7/1999 4:00:00 AM	tapi32.dll	5.00.2182.1	123.27 KB (126,224 bytes)	12/7/1999
Microsoft Corporation	c:\winnt\system32\wshnetbs.dll			4:00:00 AM	Microsoft Corporation	c:\winnt\system32\tapi32.dll	
rasadhlp.dll	5.00.2168.1	7.27 KB (7,440 bytes)	12/7/1999 4:00:00 AM	rasman.dll	5.00.2188.1	54.77 KB (56,080 bytes)	12/7/1999
Microsoft Corporation	c:\winnt\system32\rasadhlp.dll			4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rasman.dll	
winrnr.dll	5.00.2160.1	18.77 KB (19,216 bytes)	12/7/1999	rasapi32.dll	5.00.2188.1	189.77 KB (194,320 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\winrnr.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rasapi32.dll	
wshtcpip.dll	5.00.2134.1	17.27 KB (17,680 bytes)	12/7/1999	rtutils.dll	5.00.2168.1	43.77 KB (44,816 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\wshtcpip.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rtutils.dll	
msafd.dll	5.00.2153.1	54.27 KB (55,568 bytes)	12/7/1999	adslrpc.dll	5.00.2172.1	127.77 KB (130,832 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\msafd.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\adslrpc.dll	
rpcss.dll	5.00.2181.1	229.27 KB (234,768 bytes)	12/7/1999	activeds.dll	5.00.2172.1	172.77 KB (176,912 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rpcss.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\activeds.dll	
svchost.exe	5.00.2134.1	7.77 KB (7,952 bytes)	12/7/1999 4:00:00 AM	mprapi.dll	5.00.2181.1	79.27 KB (81,168 bytes)	12/7/1999
Microsoft Corporation	c:\winnt\system32\svchost.exe			4:00:00 AM	Microsoft Corporation	c:\winnt\system32\mprapi.dll	
vipl.dll	4.2.0.75	80.00 KB (81,920 bytes)	7/20/2001	iphlpapi.dll	5.00.2173.2	67.77 KB (69,392 bytes)	12/7/1999
1:28:11 PM	Giganet Incorporated	c:\winnt\system32\vipl.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\iphlpapi.dll	
gnconmgr.exe	4.2.0.75	168.05 KB (172,079 bytes)	6/5/2001	rnr20.dll	5.00.2152.1	35.77 KB (36,624 bytes)	12/7/1999
1:12:35 PM	Giganet Incorporated	c:\winnt\system32\gnconmgr.exe		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rnr20.dll	
scecli.dll	5.00.2191.1	105.27 KB (107,792 bytes)	12/7/1999	xactsrv.dll	5.00.2134.1	90.27 KB (92,432 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\scecli.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\xactsrv.dll	
atl.dll	3.00.8449	57.56 KB (58,938 bytes)	12/7/1999 4:00:00 AM	ntlsapi.dll	5.00.2134.1	6.77 KB (6,928 bytes)	12/7/1999 4:00:00 AM
Microsoft Corporation	c:\winnt\system32\atl.dll			Microsoft Corporation	c:\winnt\system32\ntlsapi.dll		
certcli.dll	5.00.2175.1	132.27 KB (135,440 bytes)	12/7/1999	wmicore.dll	5.00.2178.1	70.77 KB (72,464 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\certcli.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\wmicore.dll	
esent.dll	6.0.3939.6	1.07 MB (1,120,016 bytes)	12/7/1999	browser.dll	5.00.2142.1	48.27 KB (49,424 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\esent.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\browser.dll	
mswsock.dll	5.00.2152.1	62.27 KB (63,760 bytes)	12/7/1999	seclogon.dll	5.00.2135.1	15.77 KB (16,144 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\mswsock.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\seclogon.dll	
ntdsatq.dll	5.00.2181.1	31.27 KB (32,016 bytes)	12/7/1999	psbase.dll	5.00.2146.1	111.77 KB (114,448 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntdsatq.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\psbase.dll	
ntdsa.dll	5.00.2195.1	993.27 KB (1,017,104 bytes)	12/7/1999	cryptsvc.dll	5.00.2181.1	61.77 KB (63,248 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntdsa.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\cryptsvc.dll	
kdcsvc.dll	5.00.2181.1	133.77 KB (136,976 bytes)	12/7/1999	cryptdll.dll	5.00.2135.1	41.27 KB (42,256 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\kdcsvc.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\cryptdll.dll	
sfmapi.dll	5.00.2134.1	38.77 KB (39,696 bytes)	12/7/1999	wkssvc.dll	5.00.2181.1	95.27 KB (97,552 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\sfmapi.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\wkssvc.dll	
rassfm.dll	5.00.2168.1	21.27 KB (21,776 bytes)	12/7/1999	srvsvc.dll	5.00.2178.1	79.27 KB (81,168 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\rassfm.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\srvsvc.dll	
mpr.dll	5.00.2146.1	53.27 KB (54,544 bytes)	12/7/1999 4:00:00 AM	cfgmgr32.dll	5.00.2134.1	16.77 KB (17,168 bytes)	12/7/1999
Microsoft Corporation	c:\winnt\system32\mpr.dll			4:00:00 AM	Microsoft Corporation	c:\winnt\system32\cfgmgr32.dll	
schannel.dll	5.00.2170.1	139.77 KB (143,120 bytes)	12/7/1999	dmserver.dll	2191.1.296.2	11.77 KB (12,048 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\schannel.dll		4:00:00 AM	VERITAS Software Corp.	c:\winnt\system32\dmserver.dll	
netlogon.dll	5.00.2182.1	347.77 KB (356,112 bytes)	12/7/1999	icmp.dll	5.00.2134.1	7.27 KB (7,440 bytes)	12/7/1999 4:00:00 AM
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\netlogon.dll		Microsoft Corporation	c:\winnt\system32\icmp.dll		
msvl_0.dll	5.00.2164.1	94.77 KB (97,040 bytes)	12/7/1999	lmhsvc.dll	5.00.2134.1	9.27 KB (9,488 bytes)	12/7/1999 4:00:00 AM
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\msvl_0.dll		Microsoft Corporation	c:\winnt\system32\lmhsvc.dll		
kerberos.dll	5.00.2181.1	196.77 KB (201,488 bytes)	12/7/1999	eventlog.dll	5.00.2178.1	43.77 KB (44,816 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\kerberos.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\eventlog.dll	
msprivs.dll	5.00.2154.1	41.50 KB (42,496 bytes)	12/7/1999	ntdsapi.dll	5.00.2160.1	56.27 KB (57,616 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\msprivs.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\ntdsapi.dll	
samsrv.dll	5.00.2192.1	357.77 KB (366,352 bytes)	12/7/1999	scesrv.dll	5.00.2188.1	225.77 KB (231,184 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\samsrv.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\scesrv.dll	
lsasrv.dll	5.00.2184.1	487.77 KB (499,472 bytes)	12/7/1999	umpnpgm.dll	5.00.2182.1	86.27 KB (88,336 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\lsasrv.dll		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\umpnpgm.dll	
lsass.exe	5.00.2184.1	32.77 KB (33,552 bytes)	12/7/1999	services.exe	5.00.2134.1	86.77 KB (88,848 bytes)	12/7/1999
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\lsass.exe		4:00:00 AM	Microsoft Corporation	c:\winnt\system32\services.exe	
dhcpcsvc.dll	5.00.2153.1	88.77 KB (90,896 bytes)	12/7/1999	clbcatq.dll	1999.9.3422.14	479.27 KB (490,768 bytes)	1/16/2001
4:00:00 AM	Microsoft Corporation	c:\winnt\system32\dhcpcsvc.dll		7:06:57 AM	Microsoft Corporation	c:\winnt\system32\clbcatq.dll	

```

oleaut32.dll 2.40.4512 600.27 KB (614,672 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\oleaut32.dll
cscui.dll 5.00.2172.1 227.27 KB (232,720 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\cscui.dll
winspool.drv 5.00.2167.1 109.77 KB (112,400 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\winspool.drv
winscard.dll 5.00.2134.1 77.27 KB (79,120 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\winscard.dll
wlnotify.dll 5.00.2164.1 53.27 KB (54,544 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wlnotify.dll
cscdll.dll 5.00.2189.1 98.27 KB (100,624 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\cscdll.dll
lz32.dll 5.00.2134.1 9.77 KB (10,000 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\lz32.dll
version.dll 5.00.2134.1 15.77 KB (16,144 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\version.dll
rsabase.dll 5.00.2150.1 128.77 KB (131,856 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\rsabase.dll
mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\mscat32.dll
ole32.dll 5.00.2181.1 966.27 KB (989,456 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\ole32.dll
imagehlp.dll 5.00.2195.1 125.27 KB (128,272 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\imagehlp.dll
msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\msasn1.dll
crypt32.dll 5.131.2173.1 465.77 KB (476,944 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\crypt32.dll
wintrust.dll 5.131.2143.1 162.27 KB (166,160 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wintrust.dll
setupapi.dll 5.00.2183.1 554.27 KB (567,568 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\setupapi.dll
winmm.dll 5.00.2161.1 184.77 KB (189,200 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\winmm.dll
comctl32.dll 5.81 540.27 KB (553,232 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\comctl32.dll
shlwapi.dll 5.00.2920.0000 282.77 KB (289,552 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\shlwapi.dll
shell32.dll 5.00.2920.0000 2.24 MB (2,352,400 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\shell32.dll
msgina.dll 5.00.2191.1 309.77 KB (317,200 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\msgina.dll
winsta.dll 5.00.2134.1 36.27 KB (37,136 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\winsta.dll
wsock32.dll 5.00.2152.1 21.27 KB (21,776 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wsock32.dll
dnsapi.dll 5.00.2181.1 129.77 KB (132,880 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\dnsapi.dll
wldap32.dll 5.00.2168.1 155.77 KB (159,504 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\wldap32.dll
ws2help.dll 5.00.2134.1 17.77 KB (18,192 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\ws2help.dll
ws2_32.dll 5.00.2134.1 69.77 KB (71,440 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\ws2_32.dll
samlib.dll 5.00.2160.1 46.27 KB (47,376 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\samlib.dll
netrap.dll 5.00.2134.1 11.27 KB (11,536 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\netrap.dll

```

```

netapi32.dll 5.00.2194.1 302.77 KB (310,032 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\netapi32.dll
profmap.dll 5.00.2181.1 29.27 KB (29,968 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\profmap.dll
secur32.dll 5.00.2154.1 46.77 KB (47,888 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\secur32.dll
sfc.dll 5.00.2164.1 84.27 KB (86,288 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\sfc.dll
nddeapi.dll 5.00.2137.1 15.27 KB (15,632 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\nddeapi.dll
userenv.dll 5.00.2185.1 361.27 KB (369,936 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\userenv.dll
msvcrt.dll 6.10.8637.0 288.09 KB (295,000 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\msvcrt.dll
winlogon.exe 5.00.2182.1 173.27 KB (177,424 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\winlogon.exe
rpcrt4.dll 5.00.2193.1 434.27 KB (444,688 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\rpcrt4.dll
advapi32.dll 5.00.2191.1 349.27 KB (357,648 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\advapi32.dll
gdi32.dll 5.00.2180.1 228.77 KB (234,256 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\gdi32.dll
kernel32.dll 5.00.2191.1 715.27 KB (732,432 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\kernel32.dll
user32.dll 5.00.2180.1 393.27 KB (402,704 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\user32.dll
winsrv.dll 5.00.2180.1 246.27 KB (252,176 bytes) 11/30/1999
3:39:54 PM Microsoft Corporation c:\winnt\system32\winsrv.dll
basesrv.dll 5.00.2191.1 35.27 KB (36,112 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\basesrv.dll
csrssrv.dll 5.00.2137.1 33.77 KB (34,576 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\csrssrv.dll
csrss.exe 5.00.2137.1 5.27 KB (5,392 bytes) 12/7/1999 4:00:00 AM
Microsoft Corporation c:\winnt\system32\csrss.exe
sfcfiles.dll 5.00.2195.1 973.27 KB (996,624 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\sfcfiles.dll
ntdll.dll 5.00.2163.1 469.77 KB (481,040 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\ntdll.dll
smss.exe 5.00.2170.1 44.27 KB (45,328 bytes) 12/7/1999
4:00:00 AM Microsoft Corporation c:\winnt\system32\smss.exe

```

[Services]

Display Name	Name	State	Start	Mode	Service Type	Path	Error
Control Start	Name Tag ID						
Alerter	Alerter	Stopped	Disabled	Share	Process		
Application Management	AppMgmt	Stopped	Manual	Share	Process		
Computer Browser	Browser	Running	Auto	Share	Process		
Indexing Service	cisvc	Stopped	Manual	Share	Process		
ClipBook	ClipSrv	Stopped	Manual	Own	Process		
Distributed File System	Dfs	Stopped	Disabled	Own	Process		
DHCP Client	Dhcp	Stopped	Disabled	Share	Process		

Logical Disk Manager Administrative Service dmadm Stopped Manual Share
 Process c:\winnt\system32\dmadmin.exe /com Normal LocalSystem 0
 Logical Disk Manager dmsrver Running Auto Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 DNS Client Dnscache Stopped Disabled Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 Event Log Eventlog Running Auto Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 COM+ Event System EventSystem Running Manual Share Process
 c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 Fax Service Fax Stopped Manual Own Process
 c:\winnt\system32\faxsvc.exe Normal LocalSystem 0
 cLAN Connection Manager GniConMgr Running Auto Own Process
 c:\winnt\system32\gnconmgr.exe Normal LocalSystem 0
 IIS Admin Service IISADMIN Running Auto Share Process
 c:\winnt\system32\inetrv\inetinfo.exe Normal LocalSystem 0
 Intersite Messaging IsmServ Stopped Disabled Own Process
 c:\winnt\system32\ismserv.exe Normal LocalSystem 0
 Kerberos Key Distribution Center kdc Stopped Disabled Share
 Process c:\winnt\system32\lsass.exe Normal LocalSystem 0
 Server lanmanserver Running Auto Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 Workstation lanmanworkstation Running Auto Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 License Logging Service LicenseService Stopped Disabled Own
 Process c:\winnt\system32\llssrv.exe Normal LocalSystem 0
 TCP/IP NetBIOS Helper Service LmHosts Running Auto Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 Messenger Messenger Stopped Disabled Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 NetMeeting Remote Desktop Sharing mnmsrvc Stopped Manual Own Process
 c:\winnt\system32\mnmsrvc.exe Normal LocalSystem 0
 Distributed Transaction Coordinator MSDTC Running Auto Own Process
 c:\winnt\system32\msdtc.exe Normal LocalSystem 0
 Windows Installer MSI Server Stopped Manual Share Process
 c:\winnt\system32\msiexec.exe /v Normal LocalSystem 0
 Network DDE NetDDE Stopped Manual Share Process
 c:\winnt\system32\netdde.exe Normal LocalSystem 0
 Network DDE DSDM NetDDEdsdm Stopped Manual Share Process
 c:\winnt\system32\netdde.exe Normal LocalSystem 0
 Net Logon Netlogon Stopped Manual Share Process
 c:\winnt\system32\lsass.exe Normal LocalSystem 0
 Network Connections Netman Running Manual Share Process
 c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 File Replication NtFrs Stopped Manual Own Process
 c:\winnt\system32\ntfrs.exe Ignore LocalSystem 0
 NT LM Security Support Provider NtLmSsp Stopped Manual Share Process
 c:\winnt\system32\lsass.exe Normal LocalSystem 0
 Removable Storage NtmsSvc Stopped Disabled Share Process
 c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 Plug and Play PlugPlay Running Auto Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 IPSEC Policy Agent PolicyAgent Stopped Disabled Share Process
 c:\winnt\system32\lsass.exe Normal LocalSystem 0

Protected Storage ProtectedStorage Running Auto Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 Remote Access Auto Connection Manager RasAuto Stopped Manual Share Process
 c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 Remote Access Connection Manager RasMan Stopped Manual Share Process
 c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 Routing and Remote Access RemoteAccess Stopped Disabled Share
 Process c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 Remote Registry Service RemoteRegistry Running Auto Own Process
 c:\winnt\system32\regsvc.exe Normal LocalSystem 0
 Remote Procedure Call (RPC) Locator RpcLocator Stopped Manual Own
 Process c:\winnt\system32\locator.exe Normal LocalSystem 0
 Remote Procedure Call (RPC) RpcSs Running Auto Share Process
 c:\winnt\system32\svchost -k rpcss Normal LocalSystem 0
 QoS RSVP RSVP Running Manual Own Process
 c:\winnt\system32\rsvp.exe -s Normal LocalSystem 0
 Security Accounts Manager SamSs Running Auto Share Process
 c:\winnt\system32\lsass.exe Normal LocalSystem 0
 Smart Card Helper SCardDrv Stopped Manual Share Process
 c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0
 Smart Card SCardSvr Stopped Manual Share Process
 c:\winnt\system32\scardsvr.exe Ignore LocalSystem 0
 Task Scheduler Schedule Running Auto Share Process
 c:\winnt\system32\mstask.exe Normal LocalSystem 0
 RunAs Service seclogon Running Auto Share Process
 c:\winnt\system32\services.exe Ignore LocalSystem 0
 System Event Notification SENS Running Auto Share Process
 c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 Internet Connection Sharing SharedAccess Stopped Manual Share Process
 c:\winnt\system32\svchost.exe -k netsvcs Normal LocalSystem 0
 Print Spooler Spooler Stopped Disabled Own Process
 c:\winnt\system32\spoolsv.exe Normal LocalSystem 0
 Performance Logs and Alerts SysmonLog Stopped Manual Own Process
 c:\winnt\system32\smlogsvc.exe Normal LocalSystem 0
 Telephony Tapisrv Stopped Disabled Share Process
 c:\winnt\system32\svchost.exe -k tapisrv Normal LocalSystem 0
 Terminal Services TermService Stopped Disabled Own Process
 c:\winnt\system32\termsrv.exe Normal LocalSystem 0
 Telnet TlntSvr Stopped Manual Own Process c:\winnt\system32\tlntsvr.exe
 Normal LocalSystem 0
 Distributed Link Tracking Server TrkSvr Stopped Disabled Share
 Process c:\winnt\system32\services.exe Normal LocalSystem 0
 Distributed Link Tracking Client TrkWks Stopped Disabled Share
 Process c:\winnt\system32\services.exe Normal LocalSystem 0
 Uninterruptible Power Supply UPS Stopped Manual Own Process
 c:\winnt\system32\ups.exe Normal LocalSystem 0
 Utility Manager UtilMan Stopped Manual Own Process
 c:\winnt\system32\utilman.exe Normal LocalSystem 0
 Windows Time W32Time Stopped Manual Share Process
 c:\winnt\system32\services.exe Normal LocalSystem 0
 World Wide Web Publishing Service W3SVC Stopped Manual Share Process
 c:\winnt\system32\inetrv\inetinfo.exe Normal LocalSystem 0

```

Windows Management Instrumentation WinMgmt Running Auto Own Process
c:\winnt\system32\wbem\winmgmt.exe Ignore LocalSystem 0
Windows Management Instrumentation Driver Extensions Wmi Running Manual
Share Process c:\winnt\system32\services.exe Normal
LocalSystem 0

```

[Program Groups]

Group Name	Name	User Name
Accessories	Default User:Accessories	Default User
Accessories\Accessibility	Default User:Accessories\Accessibility	Default User
Accessories\Entertainment	Default User:Accessories\Entertainment	Default User
Accessories\System Tools	Default User:Accessories\System Tools	Default User
Startup	Default User:Startup	Default User
Accessories	All Users:Accessories	All Users
Accessories\Communications	All Users:Accessories\Communications	All Users
Accessories\Entertainment	All Users:Accessories\Entertainment	All Users
Accessories\System Tools	All Users:Accessories\System Tools	All Users
Administrative Tools	All Users:Administrative Tools	All Users
GigaNet	All Users:GigaNet	All Users
Microsoft SQL Server	All Users:Microsoft SQL Server	All Users
Startup	All Users:Startup	All Users
Accessories	CLIENT1\Administrator:Accessories	CLIENT1\Administrator
Accessories\Accessibility	CLIENT1\Administrator:Accessories\Accessibility	CLIENT1\Administrator
Accessories\Entertainment	CLIENT1\Administrator:Accessories\Entertainment	CLIENT1\Administrator
Accessories\System Tools	CLIENT1\Administrator:Accessories\System Tools	CLIENT1\Administrator
Administrative Tools	CLIENT1\Administrator:Administrative Tools	CLIENT1\Administrator
Startup	CLIENT1\Administrator:Startup	CLIENT1\Administrator

[Startup Programs]

Program Command	User Name	Location
Start rhostsvr	c:\rhost\rhostsvr.exe	-d\rhost -crhosthdlr.exe
	CLIENT1\Administrator	Startup

[OLE Registration]

Object	Local Server
Sound (OLE2)	sndrec32.exe
Media Clip	mplay32.exe
Video Clip	mplay32.exe /avi
MIDI Sequence	mplay32.exe /mid
Sound	Not Available
Media Clip	Not Available
Image Document	"C:\Program Files\Windows NT\Accessories\ImageVue\KodakImg.exe"
WordPad Document	"%ProgramFiles%\Windows NT\Accessories\WORDPAD.EXE"

```

Windows Media Services DRM Storage object Not Available
Bitmap Image mspaint.exe

```

[Internet Explorer 5]

[Following are sub-categories of this main category]

[Summary]

Item	Value
Version	5.00.2920.0000
Build	52920
Product ID	51876-270-6563967-05450
Application Path	C:\Program Files\Internet Explorer
Language	English (United States)
Active Printer	Not Available

Cipher Strength	56-bit
Content Advisor	Disabled
IEAK Install	No

[File Versions]

File	Version	Size	Date	Path	Company
advapi32.dll	5.0.2191.1	349 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
advpack.dll	5.0.2920.0	87 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
browseui.dll	5.0.2920.0	35 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
browseui.dll	5.0.2920.0	793 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
ckcnv.exe	5.0.2189.1	9 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
comctl32.dll	5.81.2920.0	540 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
crypt32.dll	5.131.2173.1	466 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
ehnsig.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iemigrat.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iesetup.dll	5.0.2920.0	57 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
ieexplore.exe	5.0.2920.0	59 KB	12/7/1999 5:00:00 AM	C:\Program Files\Internet Explorer	Microsoft Corporation
imagehlp.dll	5.0.2195.1	125 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
imgghelp.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
inseng.dll	5.0.2920.0	72 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
jobexec.dll	5.0.0.147 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation	
jscrip.dll	5.1.0.4615	476 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation
jsproxy.dll	5.0.2920.0	13 KB	12/7/1999 5:00:00 AM	C:\WINNT\system32	Microsoft Corporation

```

msahtml.dll <File Missing> Not Available Not Available Not Available
  Not Available
mshtml.dll 5.0.2920.0 2302 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
msjava.dll 5.0.3234.0 918 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
msoss.dll <File Missing> Not Available Not Available Not Available
  Not Available
msxml.dll 5.0.2920.0 521 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
occache.dll 5.0.2920.0 86 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
ole32.dll 5.0.2181.1 966 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
oleaut32.dll 2.40.4512.1 600 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
olepro32.dll 5.0.4512.1 160 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
rsabase.dll 5.0.2150.1 129 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
rsaenh.dll <File Missing> Not Available Not Available Not Available
  Not Available
rsapi32.dll <File Missing> Not Available Not Available Not Available
  Not Available
rsasig.dll <File Missing> Not Available Not Available Not Available
  Not Available
schannel.dll 5.0.2170.0 140 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
shdoc401.dll <File Missing> Not Available Not Available Not Available
  Not Available
shdocvw.dll 5.0.2920.0 1078 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
shell32.dll 5.0.2920.0 2297 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
shlwapi.dll 5.0.2920.0 283 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
url.dll 5.0.2920.0 82 KB 12/7/1999 5:00:00 AM C:\WINNT\system32
  Microsoft Corporation
urlmon.dll 5.0.2920.0 427 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
vbscript.dll 5.1.0.4615 428 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
webcheck.dll 5.0.2920.0 252 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
win.com 5.0.2134.1 24 KB 12/7/1999 5:00:00 AM C:\WINNT\system32
  Microsoft Corporation
wininet.dll 5.0.2920.0 457 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
winsock.dll 3.10.0.103 3 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
wintrust.dll 5.131.2143.1 162 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
wsock.vxd <File Missing> Not Available Not Available Not Available
  Not Available
wsock32.dll 5.0.2152.1 21 KB 12/7/1999 5:00:00 AM
  C:\WINNT\system32 Microsoft Corporation
wsock32n.dll <File Missing> Not Available Not Available Not Available
  Not Available

```

```

[Connectivity]
Item Value
Connection Preference Never dial
EnableHttp1.1 1
ProxyHttp1.1 0

LAN Settings
AutoConfigProxy Not Available
AutoProxyDetectMode Enabled
AutoConfigURL
Proxy Disabled
ProxyServer
ProxyOverride

[Cache]

[ Following are sub-categories of this main category ]

[Summary]
Item Value
Page Refresh Type Automatic
Temporary Internet Files Folder C:\Documents and
Settings\Administrator\Local Settings\Temporary Internet Files
Total Disk Space 17351 MB
Available Disk Space 15554 MB
Maximum Cache Size 542 MB
Available Cache Size 542 MB

[List of Objects]
Program File Status CodeBase
No cached object information available

[Content]

[ Following are sub-categories of this main category ]

[Summary]
Item Value
Content Advisor Disabled

[Personal Certificates]
Issued To Issued By Validity Signature Algorithm
Administrator Administrator 1/16/2001 to 12/23/2100 sha1RSA

[Other People Certificates]
Issued To Issued By Validity Signature Algorithm
No other people certificate information available

[Publishers]
Name
No publisher information available

```


[Security]

Zone Security Level
Local intranet Medium-low
Trusted sites Low
Internet Medium
Restricted sites High

Component Services Configuration

Component Services Configuration

COM+ Component TPCC.AllTxns Settings:

Transactions not supported
Enable object pooling
Minimum pool size 63
Maximum pool size 63
Creation timeout 120,000
Enable object construction
Enable just in time activation
Concurrency required

Internet Information Server Registry Parameters

\registry\machine\system\currentcontrolset\services\inetinfo
Parameters

ListenBackLog = REG_DWORD 0x00000019
DispatchEntries = REG_MULTI_SZ "LDAPSVC"
BandwidthLevel = REG_DWORD 0xffffffff
ThreadTimeout = REG_DWORD 0x00015180
DisableMemoryCache = REG_DWORD 0x00000001
MemoryCacheSize = REG_DWORD 0x00000000
PoolThreadLimit = REG_DWORD 0x000000c8
ObjectCacheTTL = REG_DWORD 0xffffffff

Performance

Library = infoctrs.dll
Open = OpenINFOPerformanceData
Close = CloseINFOPerformanceData
Collect = CollectINFOPerformanceData
Last Counter = REG_DWORD 0x00000842
Last Help = REG_DWORD 0x00000843
First Counter = REG_DWORD 0x00000802
First Help = REG_DWORD 0x00000803
Library Validation Code = REG_BINARY 0x00000010 0x03d6ec42
0x01c07fce 0x00002510 0x00000000
WbemAdapFileTime = REG_BINARY 0x00000008 0x964e6000 0x01bf40aa
WbemAdapFileSize = REG_DWORD 0x00002510
WbemAdapStatus = REG_DWORD 0x00000000

MSSQL Server Registry Parameters

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1431,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1432,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1433,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1434,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1435,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1421,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1422,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1423,0

\registry\machine\software\microsoft\mssqlserver\client\connectto
DSQUERY = DBMSSOCN
DBSERVER = DBNETLIB, via:mv-1e030p1,1424,0

Start World Wide Web Server Registry Parameters

\registry\machine\system\currentcontrolset\services\w3svc
Type = REG_DWORD 0x00000020
Start = REG_DWORD 0x00000003
ErrorControl = REG_DWORD 0x00000001
ImagePath = REG_EXPAND_SZ C:\WINNT\System32\inet_srv\inetinfo.exe
DisplayName = World Wide Web Publishing Service
DependOnService = REG_MULTI_SZ "IISADMIN"
DependOnGroup = REG_MULTI_SZ
ObjectName = LocalSystem
Description = Provides Web connectivity and administration through the
Internet Information Services snap-in.
ASP
NOTE = This is for backward compatibility only.
Parameters
Parameters
MajorVersion = REG_DWORD 0x00000005

```

MinorVersion = REG_DWORD 0x00000000
InstallPath = C:\WINNT\System32\inetsrv
CertMapList = C:\WINNT\System32\inetsrv\iiscrmap.dll
AccessDeniedMessage = Error: Access is Denied.
Filter DLLs =
LogFileDirectory = C:\WINNT\System32\LogFiles
AcceptExOutStanding = REG_DWORD 0x00000028
ADCLaunch
    AdvancedDataFactory
    RDSServer.DataFactory
Script Map
Virtual Roots
    / = c:\inetpub\wwwroot,,207
    /Scripts = c:\inetpub\scripts,,204
    /IISHelp = c:\winnt\help\iishelp,,1
    /IISAdmin = C:\WINNT\System32\inetsrv\iisadmin,,201
    /IISSamples = c:\inetpub\iissamples,,201
    /MSADC = c:\program files\common files\system\msadc,,205
    /Printers = C:\WINNT\web\printers,,201
Performance
    Library = w3ctrs.dll
    Open = OpenW3PerformanceData
    Close = CloseW3PerformanceData
    Collect = CollectW3PerformanceData
    Last Counter = REG_DWORD 0x000008e6
    Last Help = REG_DWORD 0x000008e7
    First Counter = REG_DWORD 0x00000844
    First Help = REG_DWORD 0x00000845
    Library Validation Code = REG_BINARY 0x00000010 0xc18a727c
0x01c026d2 0x00003d10 0x00000000
    WbemAdapFileType = REG_BINARY 0x00000008 0x964e6000 0x01bf40aa
    WbemAdapFileSize = REG_DWORD 0x00003d10

```

```

WbemAdapStatus = REG_DWORD 0x00000000
Security [17 1]
    Security = REG_BINARY 0x000000b8 0x80140001 0x000000a0 0x000000ac
0x00000014 0x00000030 0x001c0002 0x00000001 0x00148002 0x000f01ff
0x00000101 0x01000000 0x00000000 0x00700002 0x00000004 0x00180000
0x000201fd 0x00000101 0x05000000 \
    0x00000012 0x006f0074 0x001c0000 0x000f01ff 0x00000201
0x05000000 0x00000020 0x00000220 0x00730072 0x00180000 0x0002018d
0x00000101 0x05000000 0x0000000b 0x00000220 0x001c0000 0x000201fd
0x00000201 0x05000000 0x00000020 \
    0x00000223 0x00730072 0x00000101 0x05000000 0x00000012
0x00000101 0x05000000 0x00000012
Enum
    0 = Root\LEGACY_W3SVC\0000
Count = REG_DWORD 0x00000001
NextInstance = REG_DWORD 0x00000001

```

TPCC Application Settings

```

\registry\machine\software\unisis
    TPCC
        MAXTERMS = 20000
        DELIVERYTHREADS = 10
        SERVERNAME = DBSERVER
        DATABASE = tpcc

```

Appendix D - RTE Code

Admin Environment

```
if '%1'==' ' goto usage
if '%2'==' ' goto usage
if '%3'==' ' goto usage

:paramok

set WEBCHECKWIDS=1
set WEBDIAGLEVEL=4
set WEBEVENTLOG=0
set WEBEVENTHOST=
set WEBCHECKLEVEL=2

c:\webdriver\webadmin.exe -cweb%1.cfg -m%2 -d%3 -s160
if %ERRORLEVEL% NEQ 0 pause

goto end

:usage
@ECHO You must supply the following parameters:
@ECHO "webnnc.cmd <cfg file suffix> <min driver #> <max driver #>"
pause

:end
```

Profiles used for Performance Run

Web11520.cfg

```
//
// Common Driver Configuration
//
INITBASEPORT 4300
INITSYNCMAX 4
INITPAUSE 1
INITRSCALE 1500
INITTSCALE 100
INITRWID 1,11520
INITFIXEDWID 1
INITCCLAST 208
INITCCID 208
INITCITEMID 208
//
//Configuration Driver 1
//
1 INITIPADDR 192.168.93.11
1 INITIISADDR 192.1.01.1
```

```
1 INITIISPORT 80
1 INITBROWSERS 800
1 INITMYWID 1,80
//
//Configuration Driver 2
//
2 INITIPADDR 192.168.93.11
2 INITIISADDR 192.1.02.1
2 INITIISPORT 80
2 INITBROWSERS 800
2 INITMYWID 81,160
//
//Configuration Driver 3
//
3 INITIPADDR 192.168.93.11
3 INITIISADDR 192.1.03.1
3 INITIISPORT 80
3 INITBROWSERS 800
3 INITMYWID 161,240
//
//Configuration Driver 4
//
4 INITIPADDR 192.168.93.11
4 INITIISADDR 192.1.04.1
4 INITIISPORT 80
4 INITBROWSERS 800
4 INITMYWID 241,320
//
//Configuration Driver 5
//
5 INITIPADDR 192.168.93.11
5 INITIISADDR 192.1.05.1
5 INITIISPORT 80
5 INITBROWSERS 800
5 INITMYWID 321,400
//
//Configuration Driver 6
//
6 INITIPADDR 192.168.93.11
6 INITIISADDR 192.1.06.1
6 INITIISPORT 80
6 INITBROWSERS 800
6 INITMYWID 401,480
//
//Configuration Driver 7
//
7 INITIPADDR 192.168.93.11
7 INITIISADDR 192.1.07.1
7 INITIISPORT 80
7 INITBROWSERS 800
7 INITMYWID 481,560
//
//Configuration Driver 8
//
```

```

8 INITIPADDR 192.168.93.11
8 INITIISADDR 192.1.08.1
8 INITIISPORT 80
8 INITBROWSERS 800
8 INITMYWID 561,640
//
//Configuration Driver 9
//
9 INITIPADDR 192.168.93.12
9 INITIISADDR 192.1.11.2
9 INITIISPORT 80
9 INITBROWSERS 800
9 INITMYWID 641,720
//
//Configuration Driver 10
//
10 INITIPADDR 192.168.93.12
10 INITIISADDR 192.1.12.2
10 INITIISPORT 80
10 INITBROWSERS 800
10 INITMYWID 721,800
//
//Configuration Driver 11
//
11 INITIPADDR 192.168.93.12
11 INITIISADDR 192.1.13.2
11 INITIISPORT 80
11 INITBROWSERS 800
11 INITMYWID 801,880
//
//Configuration Driver 12
//
12 INITIPADDR 192.168.93.12
12 INITIISADDR 192.1.14.2
12 INITIISPORT 80
12 INITBROWSERS 800
12 INITMYWID 881,960
//
//Configuration Driver 13
//
13 INITIPADDR 192.168.93.12
13 INITIISADDR 192.1.15.2
13 INITIISPORT 80
13 INITBROWSERS 800
13 INITMYWID 961,1040
//
//Configuration Driver 14
//
14 INITIPADDR 192.168.93.12
14 INITIISADDR 192.1.16.2
14 INITIISPORT 80
14 INITBROWSERS 800
14 INITMYWID 1041,1120
//
//Configuration Driver 15
//
15 INITIPADDR 192.168.93.12
15 INITIISADDR 192.1.17.2
15 INITIISPORT 80

```

```

15 INITBROWSERS 800
15 INITMYWID 1121,1200
//
//Configuration Driver 16
//
16 INITIPADDR 192.168.93.12
16 INITIISADDR 192.1.18.2
16 INITIISPORT 80
16 INITBROWSERS 800
16 INITMYWID 1201,1280
//
//Configuration Driver 17
//
17 INITIPADDR 192.168.93.21
17 INITIISADDR 192.2.01.1
17 INITIISPORT 80
17 INITBROWSERS 800
17 INITMYWID 1281,1360
//
//Configuration Driver 18
//
18 INITIPADDR 192.168.93.21
18 INITIISADDR 192.2.02.1
18 INITIISPORT 80
18 INITBROWSERS 800
18 INITMYWID 1361,1440
//
//Configuration Driver 19
//
19 INITIPADDR 192.168.93.21
19 INITIISADDR 192.2.03.1
19 INITIISPORT 80
19 INITBROWSERS 800
19 INITMYWID 1441,1520
//
//Configuration Driver 20
//
20 INITIPADDR 192.168.93.21
20 INITIISADDR 192.2.04.1
20 INITIISPORT 80
20 INITBROWSERS 800
20 INITMYWID 1521,1600
//
//Configuration Driver 21
//
21 INITIPADDR 192.168.93.21
21 INITIISADDR 192.2.05.1
21 INITIISPORT 80
21 INITBROWSERS 800
21 INITMYWID 1601,1680
//
//Configuration Driver 22
//
22 INITIPADDR 192.168.93.21
22 INITIISADDR 192.2.06.1
22 INITIISPORT 80
22 INITBROWSERS 800
22 INITMYWID 1681,1760
//

```

```

//Configuration Driver 23
//
23 INITIPADDR 192.168.93.21
23 INITIISADDR 192.2.07.1
23 INITIISPORT 80
23 INITBROWSERS 800
23 INITMYWID 1761,1840
//
//Configuration Driver 24
//
24 INITIPADDR 192.168.93.21
24 INITIISADDR 192.2.08.1
24 INITIISPORT 80
24 INITBROWSERS 800
24 INITMYWID 1841,1920
//
//Configuration Driver 25
//
25 INITIPADDR 192.168.93.22
25 INITIISADDR 192.2.11.2
25 INITIISPORT 80
25 INITBROWSERS 800
25 INITMYWID 1921,2000
//
//Configuration Driver 26
//
26 INITIPADDR 192.168.93.22
26 INITIISADDR 192.2.12.2
26 INITIISPORT 80
26 INITBROWSERS 800
26 INITMYWID 2001,2080
//
//Configuration Driver 27
//
27 INITIPADDR 192.168.93.22
27 INITIISADDR 192.2.13.2
27 INITIISPORT 80
27 INITBROWSERS 800
27 INITMYWID 2081,2160
//
//Configuration Driver 28
//
28 INITIPADDR 192.168.93.22
28 INITIISADDR 192.2.14.2
28 INITIISPORT 80
28 INITBROWSERS 800
28 INITMYWID 2161,2240
//
//Configuration Driver 29
//
29 INITIPADDR 192.168.93.22
29 INITIISADDR 192.2.15.2
29 INITIISPORT 80
29 INITBROWSERS 800
29 INITMYWID 2241,2320
//
//Configuration Driver 30
//
30 INITIPADDR 192.168.93.22

```

```

30 INITIISADDR 192.2.16.2
30 INITIISPORT 80
30 INITBROWSERS 800
30 INITMYWID 2321,2400
//
//Configuration Driver 31
//
31 INITIPADDR 192.168.93.22
31 INITIISADDR 192.2.17.2
31 INITIISPORT 80
31 INITBROWSERS 800
31 INITMYWID 2401,2480
//
//Configuration Driver 32
//
32 INITIPADDR 192.168.93.22
32 INITIISADDR 192.2.18.2
32 INITIISPORT 80
32 INITBROWSERS 800
32 INITMYWID 2481,2560
//
//Configuration Driver 33
//
33 INITIPADDR 192.168.93.31
33 INITIISADDR 192.3.01.1
33 INITIISPORT 80
33 INITBROWSERS 800
33 INITMYWID 2561,2640
//
//Configuration Driver 34
//
34 INITIPADDR 192.168.93.31
34 INITIISADDR 192.3.02.1
34 INITIISPORT 80
34 INITBROWSERS 800
34 INITMYWID 2641,2720
//
//Configuration Driver 35
//
35 INITIPADDR 192.168.93.31
35 INITIISADDR 192.3.03.1
35 INITIISPORT 80
35 INITBROWSERS 800
35 INITMYWID 2721,2800
//
//Configuration Driver 36
//
36 INITIPADDR 192.168.93.31
36 INITIISADDR 192.3.04.1
36 INITIISPORT 80
36 INITBROWSERS 800
36 INITMYWID 2801,2880
//
//Configuration Driver 37
//
37 INITIPADDR 192.168.93.31
37 INITIISADDR 192.3.05.1
37 INITIISPORT 80
37 INITBROWSERS 800

```

```

37 INITMYWID 2881,2960
//
//Configuration Driver 38
//
38 INITIPADDR 192.168.93.31
38 INITIISADDR 192.3.06.1
38 INITIISPORT 80
38 INITBROWSERS 800
38 INITMYWID 2961,3040
//
//Configuration Driver 39
//
39 INITIPADDR 192.168.93.31
39 INITIISADDR 192.3.07.1
39 INITIISPORT 80
39 INITBROWSERS 800
39 INITMYWID 3041,3120
//
//Configuration Driver 40
//
40 INITIPADDR 192.168.93.31
40 INITIISADDR 192.3.08.1
40 INITIISPORT 80
40 INITBROWSERS 800
40 INITMYWID 3121,3200
//
//Configuration Driver 41
//
41 INITIPADDR 192.168.93.32
41 INITIISADDR 192.3.11.2
41 INITIISPORT 80
41 INITBROWSERS 800
41 INITMYWID 3201,3280
//
//Configuration Driver 42
//
42 INITIPADDR 192.168.93.32
42 INITIISADDR 192.3.12.2
42 INITIISPORT 80
42 INITBROWSERS 800
42 INITMYWID 3281,3360
//
//Configuration Driver 43
//
43 INITIPADDR 192.168.93.32
43 INITIISADDR 192.3.13.2
43 INITIISPORT 80
43 INITBROWSERS 800
43 INITMYWID 3361,3440
//
//Configuration Driver 44
//
44 INITIPADDR 192.168.93.32
44 INITIISADDR 192.3.14.2
44 INITIISPORT 80
44 INITBROWSERS 800
44 INITMYWID 3441,3520
//
//Configuration Driver 45

```

```

//
45 INITIPADDR 192.168.93.32
45 INITIISADDR 192.3.15.2
45 INITIISPORT 80
45 INITBROWSERS 800
45 INITMYWID 3521,3600
//
//Configuration Driver 46
//
46 INITIPADDR 192.168.93.32
46 INITIISADDR 192.3.16.2
46 INITIISPORT 80
46 INITBROWSERS 800
46 INITMYWID 3601,3680
//
//Configuration Driver 47
//
47 INITIPADDR 192.168.93.32
47 INITIISADDR 192.3.17.2
47 INITIISPORT 80
47 INITBROWSERS 800
47 INITMYWID 3681,3760
//
//Configuration Driver 48
//
48 INITIPADDR 192.168.93.32
48 INITIISADDR 192.3.18.2
48 INITIISPORT 80
48 INITBROWSERS 800
48 INITMYWID 3761,3840
//
//Configuration Driver 49
//
49 INITIPADDR 192.168.93.41
49 INITIISADDR 192.4.01.1
49 INITIISPORT 80
49 INITBROWSERS 800
49 INITMYWID 3841,3920
//
//Configuration Driver 50
//
50 INITIPADDR 192.168.93.41
50 INITIISADDR 192.4.02.1
50 INITIISPORT 80
50 INITBROWSERS 800
50 INITMYWID 3921,4000
//
//Configuration Driver 51
//
51 INITIPADDR 192.168.93.41
51 INITIISADDR 192.4.03.1
51 INITIISPORT 80
51 INITBROWSERS 800
51 INITMYWID 4001,4080
//
//Configuration Driver 52
//
52 INITIPADDR 192.168.93.41
52 INITIISADDR 192.4.04.1

```

```
52 INITIISPORT 80
52 INITBROWSERS 800
52 INITMYWID 4081,4160
//
//Configuration Driver 53
//
53 INITIPADDR 192.168.93.41
53 INITIISADDR 192.4.05.1
53 INITIISPORT 80
53 INITBROWSERS 800
53 INITMYWID 4161,4240
//
//Configuration Driver 54
//
54 INITIPADDR 192.168.93.41
54 INITIISADDR 192.4.06.1
54 INITIISPORT 80
54 INITBROWSERS 800
54 INITMYWID 4241,4320
//
//Configuration Driver 55
//
55 INITIPADDR 192.168.93.41
55 INITIISADDR 192.4.07.1
55 INITIISPORT 80
55 INITBROWSERS 800
55 INITMYWID 4321,4400
//
//Configuration Driver 56
//
56 INITIPADDR 192.168.93.41
56 INITIISADDR 192.4.08.1
56 INITIISPORT 80
56 INITBROWSERS 800
56 INITMYWID 4401,4480
//
//Configuration Driver 57
//
57 INITIPADDR 192.168.93.42
57 INITIISADDR 192.4.11.2
57 INITIISPORT 80
57 INITBROWSERS 800
57 INITMYWID 4481,4560
//
//Configuration Driver 58
//
58 INITIPADDR 192.168.93.42
58 INITIISADDR 192.4.12.2
58 INITIISPORT 80
58 INITBROWSERS 800
58 INITMYWID 4561,4640
//
//Configuration Driver 59
//
59 INITIPADDR 192.168.93.42
59 INITIISADDR 192.4.13.2
59 INITIISPORT 80
59 INITBROWSERS 800
59 INITMYWID 4641,4720
```

```
//
//Configuration Driver 60
//
60 INITIPADDR 192.168.93.42
60 INITIISADDR 192.4.14.2
60 INITIISPORT 80
60 INITBROWSERS 800
60 INITMYWID 4721,4800
//
//Configuration Driver 61
//
61 INITIPADDR 192.168.93.42
61 INITIISADDR 192.4.15.2
61 INITIISPORT 80
61 INITBROWSERS 800
61 INITMYWID 4801,4880
//
//Configuration Driver 62
//
62 INITIPADDR 192.168.93.42
62 INITIISADDR 192.4.16.2
62 INITIISPORT 80
62 INITBROWSERS 800
62 INITMYWID 4881,4960
//
//Configuration Driver 63
//
63 INITIPADDR 192.168.93.42
63 INITIISADDR 192.4.17.2
63 INITIISPORT 80
63 INITBROWSERS 800
63 INITMYWID 4961,5040
//
//Configuration Driver 64
//
64 INITIPADDR 192.168.93.42
64 INITIISADDR 192.4.18.2
64 INITIISPORT 80
64 INITBROWSERS 800
64 INITMYWID 5041,5120
//
//Configuration Driver 65
//
65 INITIPADDR 192.168.93.51
65 INITIISADDR 192.5.01.1
65 INITIISPORT 80
65 INITBROWSERS 800
65 INITMYWID 5121,5200
//
//Configuration Driver 66
//
66 INITIPADDR 192.168.93.51
66 INITIISADDR 192.5.02.1
66 INITIISPORT 80
66 INITBROWSERS 800
66 INITMYWID 5201,5280
//
//Configuration Driver 67
//
```

```

67 INITIPADDR 192.168.93.51
67 INITIISADDR 192.5.03.1
67 INITIISPORT 80
67 INITBROWSERS 800
67 INITMYWID 5281,5360
//
//Configuration Driver 68
//
68 INITIPADDR 192.168.93.51
68 INITIISADDR 192.5.04.1
68 INITIISPORT 80
68 INITBROWSERS 800
68 INITMYWID 5361,5440
//
//Configuration Driver 69
//
69 INITIPADDR 192.168.93.51
69 INITIISADDR 192.5.05.1
69 INITIISPORT 80
69 INITBROWSERS 800
69 INITMYWID 5441,5520
//
//Configuration Driver 70
//
70 INITIPADDR 192.168.93.51
70 INITIISADDR 192.5.06.1
70 INITIISPORT 80
70 INITBROWSERS 800
70 INITMYWID 5521,5600
//
//Configuration Driver 71
//
71 INITIPADDR 192.168.93.51
71 INITIISADDR 192.5.07.1
71 INITIISPORT 80
71 INITBROWSERS 800
71 INITMYWID 5601,5680
//
//Configuration Driver 72
//
72 INITIPADDR 192.168.93.51
72 INITIISADDR 192.5.08.1
72 INITIISPORT 80
72 INITBROWSERS 800
72 INITMYWID 5681,5760
//
//Configuration Driver 73
//
73 INITIPADDR 192.168.93.52
73 INITIISADDR 192.5.11.2
73 INITIISPORT 80
73 INITBROWSERS 800
73 INITMYWID 5761,5840
//
//Configuration Driver 74
//
74 INITIPADDR 192.168.93.52
74 INITIISADDR 192.5.12.2
74 INITIISPORT 80

```

```

74 INITBROWSERS 800
74 INITMYWID 5841,5920
//
//Configuration Driver 75
//
75 INITIPADDR 192.168.93.52
75 INITIISADDR 192.5.13.2
75 INITIISPORT 80
75 INITBROWSERS 800
75 INITMYWID 5921,6000
//
//Configuration Driver 76
//
76 INITIPADDR 192.168.93.52
76 INITIISADDR 192.5.14.2
76 INITIISPORT 80
76 INITBROWSERS 800
76 INITMYWID 6001,6080
//
//Configuration Driver 77
//
77 INITIPADDR 192.168.93.52
77 INITIISADDR 192.5.15.2
77 INITIISPORT 80
77 INITBROWSERS 800
77 INITMYWID 6081,6160
//
//Configuration Driver 78
//
78 INITIPADDR 192.168.93.52
78 INITIISADDR 192.5.16.2
78 INITIISPORT 80
78 INITBROWSERS 800
78 INITMYWID 6161,6240
//
//Configuration Driver 79
//
79 INITIPADDR 192.168.93.52
79 INITIISADDR 192.5.17.2
79 INITIISPORT 80
79 INITBROWSERS 800
79 INITMYWID 6241,6320
//
//Configuration Driver 80
//
80 INITIPADDR 192.168.93.52
80 INITIISADDR 192.5.18.2
80 INITIISPORT 80
80 INITBROWSERS 800
80 INITMYWID 6321,6400
//
//Configuration Driver 81
//
81 INITIPADDR 192.168.93.61
81 INITIISADDR 192.6.01.1
81 INITIISPORT 80
81 INITBROWSERS 800
81 INITMYWID 6401,6480
//

```



```

//Configuration Driver 82
//
82 INITIPADDR 192.168.93.61
82 INITIISADDR 192.6.02.1
82 INITIISPORT 80
82 INITBROWSERS 800
82 INITMYWID 6481,6560
//
//Configuration Driver 83
//
83 INITIPADDR 192.168.93.61
83 INITIISADDR 192.6.03.1
83 INITIISPORT 80
83 INITBROWSERS 800
83 INITMYWID 6561,6640
//
//Configuration Driver 84
//
84 INITIPADDR 192.168.93.61
84 INITIISADDR 192.6.04.1
84 INITIISPORT 80
84 INITBROWSERS 800
84 INITMYWID 6641,6720
//
//Configuration Driver 85
//
85 INITIPADDR 192.168.93.61
85 INITIISADDR 192.6.05.1
85 INITIISPORT 80
85 INITBROWSERS 800
85 INITMYWID 6721,6800
//
//Configuration Driver 86
//
86 INITIPADDR 192.168.93.61
86 INITIISADDR 192.6.06.1
86 INITIISPORT 80
86 INITBROWSERS 800
86 INITMYWID 6801,6880
//
//Configuration Driver 87
//
87 INITIPADDR 192.168.93.61
87 INITIISADDR 192.6.07.1
87 INITIISPORT 80
87 INITBROWSERS 800
87 INITMYWID 6881,6960
//
//Configuration Driver 88
//
88 INITIPADDR 192.168.93.61
88 INITIISADDR 192.6.08.1
88 INITIISPORT 80
88 INITBROWSERS 800
88 INITMYWID 6961,7040
//
//Configuration Driver 89
//
89 INITIPADDR 192.168.93.62

```

```

89 INITIISADDR 192.6.11.2
89 INITIISPORT 80
89 INITBROWSERS 800
89 INITMYWID 7041,7120
//
//Configuration Driver 90
//
90 INITIPADDR 192.168.93.62
90 INITIISADDR 192.6.12.2
90 INITIISPORT 80
90 INITBROWSERS 800
90 INITMYWID 7121,7200
//
//Configuration Driver 91
//
91 INITIPADDR 192.168.93.62
91 INITIISADDR 192.6.13.2
91 INITIISPORT 80
91 INITBROWSERS 800
91 INITMYWID 7201,7280
//
//Configuration Driver 92
//
92 INITIPADDR 192.168.93.62
92 INITIISADDR 192.6.14.2
92 INITIISPORT 80
92 INITBROWSERS 800
92 INITMYWID 7281,7360
//
//Configuration Driver 93
//
93 INITIPADDR 192.168.93.62
93 INITIISADDR 192.6.15.2
93 INITIISPORT 80
93 INITBROWSERS 800
93 INITMYWID 7361,7440
//
//Configuration Driver 94
//
94 INITIPADDR 192.168.93.62
94 INITIISADDR 192.6.16.2
94 INITIISPORT 80
94 INITBROWSERS 800
94 INITMYWID 7441,7520
//
//Configuration Driver 95
//
95 INITIPADDR 192.168.93.62
95 INITIISADDR 192.6.17.2
95 INITIISPORT 80
95 INITBROWSERS 800
95 INITMYWID 7521,7600
//
//Configuration Driver 96
//
96 INITIPADDR 192.168.93.62
96 INITIISADDR 192.6.18.2
96 INITIISPORT 80
96 INITBROWSERS 800

```

```
96 INITMYWID 7601,7680
//
//Configuration Driver 97
//
97 INITIPADDR 192.168.93.71
97 INITIISADDR 192.7.01.1
97 INITIISPORT 80
97 INITBROWSERS 800
97 INITMYWID 7681,7760
//
//Configuration Driver 98
//
98 INITIPADDR 192.168.93.71
98 INITIISADDR 192.7.02.1
98 INITIISPORT 80
98 INITBROWSERS 800
98 INITMYWID 7761,7840
//
//Configuration Driver 99
//
99 INITIPADDR 192.168.93.71
99 INITIISADDR 192.7.03.1
99 INITIISPORT 80
99 INITBROWSERS 800
99 INITMYWID 7841,7920
//
//Configuration Driver 100
//
100 INITIPADDR 192.168.93.71
100 INITIISADDR 192.7.04.1
100 INITIISPORT 80
100 INITBROWSERS 800
100 INITMYWID 7921,8000
//
//Configuration Driver 101
//
101 INITIPADDR 192.168.93.71
101 INITIISADDR 192.7.05.1
101 INITIISPORT 80
101 INITBROWSERS 800
101 INITMYWID 8001,8080
//
//Configuration Driver 102
//
102 INITIPADDR 192.168.93.71
102 INITIISADDR 192.7.06.1
102 INITIISPORT 80
102 INITBROWSERS 800
102 INITMYWID 8081,8160
//
//Configuration Driver 103
//
103 INITIPADDR 192.168.93.71
103 INITIISADDR 192.7.07.1
103 INITIISPORT 80
103 INITBROWSERS 800
103 INITMYWID 8161,8240
//
//Configuration Driver 104
```

```
//
104 INITIPADDR 192.168.93.71
104 INITIISADDR 192.7.08.1
104 INITIISPORT 80
104 INITBROWSERS 800
104 INITMYWID 8241,8320
//
//Configuration Driver 105
//
105 INITIPADDR 192.168.93.72
105 INITIISADDR 192.7.11.2
105 INITIISPORT 80
105 INITBROWSERS 800
105 INITMYWID 8321,8400
//
//Configuration Driver 106
//
106 INITIPADDR 192.168.93.72
106 INITIISADDR 192.7.12.2
106 INITIISPORT 80
106 INITBROWSERS 800
106 INITMYWID 8401,8480
//
//Configuration Driver 107
//
107 INITIPADDR 192.168.93.72
107 INITIISADDR 192.7.13.2
107 INITIISPORT 80
107 INITBROWSERS 800
107 INITMYWID 8481,8560
//
//Configuration Driver 108
//
108 INITIPADDR 192.168.93.72
108 INITIISADDR 192.7.14.2
108 INITIISPORT 80
108 INITBROWSERS 800
108 INITMYWID 8561,8640
//
//Configuration Driver 109
//
109 INITIPADDR 192.168.93.72
109 INITIISADDR 192.7.15.2
109 INITIISPORT 80
109 INITBROWSERS 800
109 INITMYWID 8641,8720
//
//Configuration Driver 110
//
110 INITIPADDR 192.168.93.72
110 INITIISADDR 192.7.16.2
110 INITIISPORT 80
110 INITBROWSERS 800
110 INITMYWID 8721,8800
//
//Configuration Driver 111
//
111 INITIPADDR 192.168.93.72
111 INITIISADDR 192.7.17.2
```

```
111 INITIISPORT 80
111 INITBROWSERS 800
111 INITMYWID 8801,8880
//
//Configuration Driver 112
//
112 INITIPADDR 192.168.93.72
112 INITIISADDR 192.7.18.2
112 INITIISPORT 80
112 INITBROWSERS 800
112 INITMYWID 8881,8960
//
//Configuration Driver 113
//
113 INITIPADDR 192.168.93.81
113 INITIISADDR 192.8.01.1
113 INITIISPORT 80
113 INITBROWSERS 800
113 INITMYWID 8961,9040
//
//Configuration Driver 114
//
114 INITIPADDR 192.168.93.81
114 INITIISADDR 192.8.02.1
114 INITIISPORT 80
114 INITBROWSERS 800
114 INITMYWID 9041,9120
//
//Configuration Driver 115
//
115 INITIPADDR 192.168.93.81
115 INITIISADDR 192.8.03.1
115 INITIISPORT 80
115 INITBROWSERS 800
115 INITMYWID 9121,9200
//
//Configuration Driver 116
//
116 INITIPADDR 192.168.93.81
116 INITIISADDR 192.8.04.1
116 INITIISPORT 80
116 INITBROWSERS 800
116 INITMYWID 9201,9280
//
//Configuration Driver 117
//
117 INITIPADDR 192.168.93.81
117 INITIISADDR 192.8.05.1
117 INITIISPORT 80
117 INITBROWSERS 800
117 INITMYWID 9281,9360
//
//Configuration Driver 118
//
118 INITIPADDR 192.168.93.81
118 INITIISADDR 192.8.06.1
118 INITIISPORT 80
118 INITBROWSERS 800
118 INITMYWID 9361,9440
```

```
//
//Configuration Driver 119
//
119 INITIPADDR 192.168.93.81
119 INITIISADDR 192.8.07.1
119 INITIISPORT 80
119 INITBROWSERS 800
119 INITMYWID 9441,9520
//
//Configuration Driver 120
//
120 INITIPADDR 192.168.93.81
120 INITIISADDR 192.8.08.1
120 INITIISPORT 80
120 INITBROWSERS 800
120 INITMYWID 9521,9600
//
//Configuration Driver 121
//
121 INITIPADDR 192.168.93.82
121 INITIISADDR 192.8.11.2
121 INITIISPORT 80
121 INITBROWSERS 800
121 INITMYWID 9601,9680
//
//Configuration Driver 122
//
122 INITIPADDR 192.168.93.82
122 INITIISADDR 192.8.12.2
122 INITIISPORT 80
122 INITBROWSERS 800
122 INITMYWID 9681,9760
//
//Configuration Driver 123
//
123 INITIPADDR 192.168.93.82
123 INITIISADDR 192.8.13.2
123 INITIISPORT 80
123 INITBROWSERS 800
123 INITMYWID 9761,9840
//
//Configuration Driver 124
//
124 INITIPADDR 192.168.93.82
124 INITIISADDR 192.8.14.2
124 INITIISPORT 80
124 INITBROWSERS 800
124 INITMYWID 9841,9920
//
//Configuration Driver 125
//
125 INITIPADDR 192.168.93.82
125 INITIISADDR 192.8.15.2
125 INITIISPORT 80
125 INITBROWSERS 800
125 INITMYWID 9921,10000
//
//Configuration Driver 126
//
```

```
126 INITIPADDR 192.168.93.82
126 INITIISADDR 192.8.16.2
126 INITIISPORT 80
126 INITBROWSERS 800
126 INITMYWID 10001,10080
//
//Configuration Driver 127
//
127 INITIPADDR 192.168.93.82
127 INITIISADDR 192.8.17.2
127 INITIISPORT 80
127 INITBROWSERS 800
127 INITMYWID 10081,10160
//
//Configuration Driver 128
//
128 INITIPADDR 192.168.93.82
128 INITIISADDR 192.8.18.2
128 INITIISPORT 80
128 INITBROWSERS 800
128 INITMYWID 10161,10240
//
//Configuration Driver 129
//
129 INITIPADDR 192.168.93.91
129 INITIISADDR 192.9.01.1
129 INITIISPORT 80
129 INITBROWSERS 800
129 INITMYWID 10241,10320
//
//Configuration Driver 130
//
130 INITIPADDR 192.168.93.91
130 INITIISADDR 192.9.02.1
130 INITIISPORT 80
130 INITBROWSERS 800
130 INITMYWID 10321,10400
//
//Configuration Driver 131
//
131 INITIPADDR 192.168.93.91
131 INITIISADDR 192.9.03.1
131 INITIISPORT 80
131 INITBROWSERS 800
131 INITMYWID 10401,10480
//
//Configuration Driver 132
//
132 INITIPADDR 192.168.93.91
132 INITIISADDR 192.9.04.1
132 INITIISPORT 80
132 INITBROWSERS 800
132 INITMYWID 10481,10560
//
//Configuration Driver 133
//
133 INITIPADDR 192.168.93.91
133 INITIISADDR 192.9.05.1
133 INITIISPORT 80
```

```
133 INITBROWSERS 800
133 INITMYWID 10561,10640
//
//Configuration Driver 134
//
134 INITIPADDR 192.168.93.91
134 INITIISADDR 192.9.06.1
134 INITIISPORT 80
134 INITBROWSERS 800
134 INITMYWID 10641,10720
//
//Configuration Driver 135
//
135 INITIPADDR 192.168.93.91
135 INITIISADDR 192.9.07.1
135 INITIISPORT 80
135 INITBROWSERS 800
135 INITMYWID 10721,10800
//
//Configuration Driver 136
//
136 INITIPADDR 192.168.93.91
136 INITIISADDR 192.9.08.1
136 INITIISPORT 80
136 INITBROWSERS 800
136 INITMYWID 10801,10880
//
//Configuration Driver 137
//
137 INITIPADDR 192.168.93.92
137 INITIISADDR 192.9.11.2
137 INITIISPORT 80
137 INITBROWSERS 800
137 INITMYWID 10881,10960
//
//Configuration Driver 138
//
138 INITIPADDR 192.168.93.92
138 INITIISADDR 192.9.12.2
138 INITIISPORT 80
138 INITBROWSERS 800
138 INITMYWID 10961,11040
//
//Configuration Driver 139
//
139 INITIPADDR 192.168.93.92
139 INITIISADDR 192.9.13.2
139 INITIISPORT 80
139 INITBROWSERS 800
139 INITMYWID 11041,11120
//
//Configuration Driver 140
//
140 INITIPADDR 192.168.93.92
140 INITIISADDR 192.9.14.2
140 INITIISPORT 80
140 INITBROWSERS 800
140 INITMYWID 11121,11200
//
```

```

//Configuration Driver 141
//
141 INITIPADDR 192.168.93.92
141 INITIISADDR 192.9.15.2
141 INITIISPORT 80
141 INITBROWSERS 800
141 INITMYWID 11201,11280
//
//Configuration Driver 142
//
142 INITIPADDR 192.168.93.92
142 INITIISADDR 192.9.16.2
142 INITIISPORT 80
142 INITBROWSERS 800
142 INITMYWID 11281,11360
//
//Configuration Driver 143
//
143 INITIPADDR 192.168.93.92
143 INITIISADDR 192.9.17.2
143 INITIISPORT 80
143 INITBROWSERS 800
143 INITMYWID 11361,11440
//
//Configuration Driver 144
//
144 INITIPADDR 192.168.93.92
144 INITIISADDR 192.9.18.2
144 INITIISPORT 80
144 INITBROWSERS 800
144 INITMYWID 11441,11520
//

```

```

set WEBTTPAYMENT=12030
set WEBTTDELIVERY=5060
set WEBTTORDERSTATUS=10070
set WEBTTSTOCKLEVEL=5060

```

```
webdriver.exe
```

```
goto end
```

```
:usage
```

```
@ECHO You must supply the following parameters:
```

```
@ECHO "webdriver.cmd <driver number>"
```

```
pause
```

```
:end
```

```
exit
```

Driver Environment

```
if '%1'==' ' goto usage
```

```
:paramok
```

```

set WEBDRIVerno=%1
set WEBADMBASEPORT=4300
set WEBDIAGLEVEL=2
set WEBEVENTLOG=1
set WEBEVENTHOST=
set WEBLOGLEVEL=1
set WEBSINGLETRAN=0
set WEBTPCCAUDIT=0
set WEBRTFUDGETM=110
set WEBNEWORDERPROB=4489
set WEBPAYMENTPROB=4305
set WEBORDERSTATUSPROB=402
set WEBDELIVERYPROB=402
set WEBSTOCKLEVELPROB=402
set WEBTTNEWORDER=12030

```


Appendix E - Disk Storage

TPC-C 60-Day Disk Space Requirements					
Warehouses	11,808	tpmC	141,138.44	tpmC/W	11.95
Table	Initial Rows	Data KB	Index KB	Extra 5% KB	Total With 5% KB
Warehouse	11,808	1,264	168	72	1,504
District	118,080	13,120	736	693	14,549
Customer	354,240,000	257,629,096	15,363,344	13,649,622	286,642,062
History (D)	354,240,000	19,680,000	72,056		19,752,056
Order (D)	354,240,000	10,857,936	5,477,464		16,335,400
New-Order	106,272,000	1,680,192	4,080	84,214	1,768,486
Order-Line (D)	3,542,392,472	221,399,536	468,824		221,868,360
Item	100,000	9,528	688	511	10,727
Stock	1,180,800,000	377,856,000	706,440	18,928,122	397,490,562
Totals KB		889,126,672	22,093,800	32,663,233	943,883,705
Db/Filegroup	Count	Size MB	MB Allocated	MB Loaded +5%	MB for 8 Hours
master, model & msdb		22	22	22	22
tempdb		10	10	10	10
mssql70_tpcc_root	1	10	10	10	10
mssql70_cs_fg	92	7,300	671,600	668,098	668,098
mssql70_misc_fg	92	3,500	322,000	253,663	313,135
Total Allocated MB			993,642	921,803	981,275
MB					
Dynamic Space MB	246,033	Sum of data for orders, order_line & history			
Static Space	675,729	Sum of data+index+5% - Dynamic Space			
Free Space	71,881	Total allocated space - (Dynamic & Static Spaces)			
Daily Growth	47,052	(Dynamic Space / (W * 62.5)) * tpmC			
Daily Spread	1,302	Free space - 1.5 * Daily growth (zero if negative)			
		0 SQL Server can be configured to eliminate Daily Spread			
60 Day Space MB	3,498,873	Static Space + 60 * (Daily Growth + Daily Spread)			
60 Day Space GB	3,416.87				
8 hr log GB	303.05	(need double for mirroring)			
Disk Capacity MB	33725	32.935 GB	Capacity of 36GB disks		
	16936	16.539 GB	Capacity of 18GB disks		
Space Usage	GB Needed	Disks Priced	GB Priced		
60-day space DB	3416.87 GB	919	15199.34 GB	18GB drives	
		0	0.00 GB	36GB drives	
Total DB		919	15199.34 GB		
8-hr log+mirror	606.09 GB	24	790.44 GB	36GB drives	
OS, SQL Server	11.00 GB	2	34.00 GB	18GB drive	
Total space	4033.96 GB	945	16023.78 GB	15.65 TB	

TPC-C 60-Day Dynamic Table Growth Rates for 8 Hours						141,138.44 tpmC
Tables	Initial (KB)	Final (KB)	Change(KB)	Unused (KB)	KB / New-Order	8-Hr MB
History	20,169,328	22,250,320	2,080,992	224	0.0594	23,627.26
Orders	17,301,696	22,296,440	4,994,744	968	0.1292	25,441.19
Order_line	226,244,320	250,426,952	24,182,632	3,608	0.6253	262,313.34
Dynamic	263,715,344	294,973,712	31,258,368	4,800	0.8139	311,381.79
New_order	2,577,264	2,801,960	224,696	108,936	0.0058	2,901.27
Static						
Log	5,668,486	183,821,887	178,153,400		4.6069	310,320.32 303.047 GB
SUM(d_next_o_id)	358,000,695	396,671,888	38,671,193			

Appendix F - Third-Party Price Quotations

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
<http://www.microsoft.com/>

Microsoft

July 24, 2001

Unisys Corp.
Glenn Weeks
MS237
25725 Jeronimo Road
Mission Viejo, CA 92691

Glenn:

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-C V5.0 benchmark testing.

All pricing shown is in US Dollars (\$).

Part Number	Description	Unit Price	Quantity	Price
810-00846	SQL Server 2000 Enterprise Edition <i>Per processor licensing</i> <i>Discount schedule: Open Program Level C</i>	\$ 16,541	32	\$ 529,312
C11-00821	Windows 2000 Server <i>Server license only - No CALs</i> <i>Discount schedule: Open Program - No Level</i>	\$ 738	1	\$ 738
048-00317	Visual C++ Professional 6.0 Win32	\$ 549	1	\$ 549
	3-year maintenance for above software	\$ 2,095	1	\$ 6,285

All products are currently orderable through Microsoft's normal distribution channels.

This quote is valid for the next 90 days.

If we can be of any further assistance, please contact Jamie Reding at (425) 703-0510 or jamiere@microsoft.com.

Reference ID: Puxfr0124078640

Please include this Reference ID in any correspondence regarding this price quote.



Advanced Systems Division

Quote # DM010801-UN007

Aug 1, 2001

To: Unisys Corporation TPC-C

Qty	Prod #	Description	Unit Price	Net Price
ES7000 Server				
1	ES73251-GDN	SYS: ES7000, 2 Domain, 32P, 8 GB, 2 MSUs, 2 DIBs	\$521,333.00	\$521,333.00
3	RAM7-16G	MEM: 16GB, 1 GB DIMMs	\$81,600.00	\$244,800.00
1	RAM7-8GB	MEM: 8GB, 1 GB DIMMs	\$43,520.00	\$43,520.00
2	MSU2000-MBD	BRD: Memory Carrier Board, 0 Mem.	\$10,880.00	\$21,760.00
6	PCI2053-12	IO: Module, PCI Enclosure w/ BID Motherboard	\$2,992.00	\$17,952.00
10	PCI3-MOD	IO: Module, PCI Enclosure 3.3V, 4 Card	\$363.00	\$3,630.00
9	PCI6221-SW	CTRL: Fibre Channel HBA, 2-Port, 64-bit PCI	\$2,063.00	\$18,567.00
2	PCI10001-CXG	CTRL: cLAN, PCI Host Adapter & Cable	\$586.00	\$1,172.00
1	PCI7602-P64	CTRL: 2Ch SCSI, U160, PCI	\$442.00	\$442.00
2	HDM18110-CX1	DISK: 18GB, 10K SCSI LVD, SCA	\$568.00	\$1,136.00
1	ES70001-UIF	I/F: Monitor, 15-inch Color, Kybrd, Mse & Cable	\$272.00	\$272.00
1	WID7323-CGS	MAINT: 3-Yr. Cmprhnsv-Gold Svc Wrrnty Upgrd	\$39,038.00	\$39,038.00
1	ESS700030-N	SYS MGT: ES7000 Value Add Software	\$4,533.00	\$4,533.00
1	LED320032-EN	O/S: Microsoft Windows Datacenter Lmtd Edtn, 1-32P	\$90,667.00	\$90,667.00
3	WM320032-EN	O/S: Windows Datacenter Annual Update, 1-32P	\$26,293.00	\$78,879.00
ESM7700 Storage				
1011	ESM18201-F96	DISK: 18GB Drive, 10K FC, SCA + 10% spares	\$615.00	\$621,765.00
27	ESM36202-F94	DISK: 36GB Drive, 10K FC, SCA + 10% spares	\$928.00	\$25,056.00
72	ESM7200-DAE	DAE: Expansion Enclsr, w/ 0 Disk	\$4,087.00	\$294,264.00
24	ESM7700-DPE	DPE: 2 RAID Cntrl, 512MB Cache ea., 0 Disk	\$29,361.00	\$704,664.00
1	ESM7700-W2K	SYS MGT: Storage Agent, Monitor, Call Home & ATF	\$1,661.00	\$1,661.00
18	CBL135-10	CBL: FC, DB9->DB9 Conn's, 10m	\$193.00	\$3,474.00
24	SFR9-PWR	PWR: Distribution Strip, 220V	\$295.00	\$7,080.00
24	USE1936-LC6	CBL: Power, U.S. (Domestic), L6-20P	\$56.00	\$1,344.00
12	RM361934-OFE	CAB: 36U x 19" x 34" Cabinet, e-@ction	\$810.00	\$9,720.00
12	RM3619-RDR	DOOR: 36U x 19", Rear	\$277.00	\$3,324.00
12	RM3634-SDS	PNL: 36U x 34" Side Skins, L&R	\$221.00	\$2,652.00
ES2024 Servers				
9	ES202141-GZN	SYS: ES2024 Tower, w/ 0 Proc., 0MB Mem	\$1,547.00	\$13,923.00
18	CPU31000-256	PROC:1x1GHz Pentium III/256KB Cache	\$884.00	\$15,912.00
9	VRM3-83	ACC: Voltage Regulator	\$29.00	\$261.00
18	DIM13368-256	MEM: 256 MB SDRAM PC133ECC Memory	\$221.00	\$3,978.00
9	ES2024-18G	DISK: 18GB SCSI 3.5 Internal	\$553.00	\$4,977.00
9	PCI10001-CXG	CTRL: cLAN, PCI Host Adapter & Cable	\$586.00	\$5,274.00
9	WI2003-CGS	MAINT: 3 Yr. Cmprhnsv-Gold Svc Warranty Upgrade	\$1,326.00	\$11,934.00
9	EVG2100-P	MONITOR:15-inch Color	\$221.00	\$1,989.00
Network				
2	SWT50082-CXG	SWITCH: cLAN, 8-port, 1.25Gbit	\$5,157.00	\$10,314.00



Advanced Systems Division

Quote # DM010801-UN007

Aug 1, 2001

To: Unisys Corporation TPC-C

Qty	Prod #	Description	Unit Price	Net Price
-----	--------	-------------	------------	-----------

1		Large Purchase Cash Prepay Discount	-\$283,127.00	-\$283,127.00
---	--	-------------------------------------	---------------	---------------

Prices may vary when items are purchased separately.
 Disks come with a 3 year return-to-factory warranty,
 7 day replenishment. Quote valid for 90 days.

TOTAL

\$2,548,140.00