UNISYS

TPC Benchmark™ C Full Disclosure Report

Unisys e-@ction Enterprise Server ES5085R

using

Microsoft SQL Server 2000 Enterprise Edition

on

Microsoft Windows 2000 Datacenter Server

Second Edition
January 18th 2001

Unisys Part Number 4500 5162-100

Second Edition – January 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.

Printed in USA, January 2001.

Unisys Corporation Part Number: 4500 5162-100

Page Status

Page	Issue
i through xii	-000
0-1 through 0-3	-000
0-4	Blank
1-1 through 1-1	-000
1-2	Blank
2-1 through 2-2	-000
3-1 through 3-4	-000
4-1 through 4-8	-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-51	-000
B-52	Blank
C-1 through C-79	-000
C-80	Blank
D-1 through D-5	-000
D-6	Blank
E-1 through E-2	-000
F-1 through F-5	-000
F-6	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.

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark C (TPC-C) conducted on the Unisys Corporation e-@ction Enterprise Server ES5085R . The operating system on the server was Microsoft Windows 2000 Datacenter Server. 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 (five year capital cost per measured tpmC), and the availability date are reported as required by the benchmark specification.

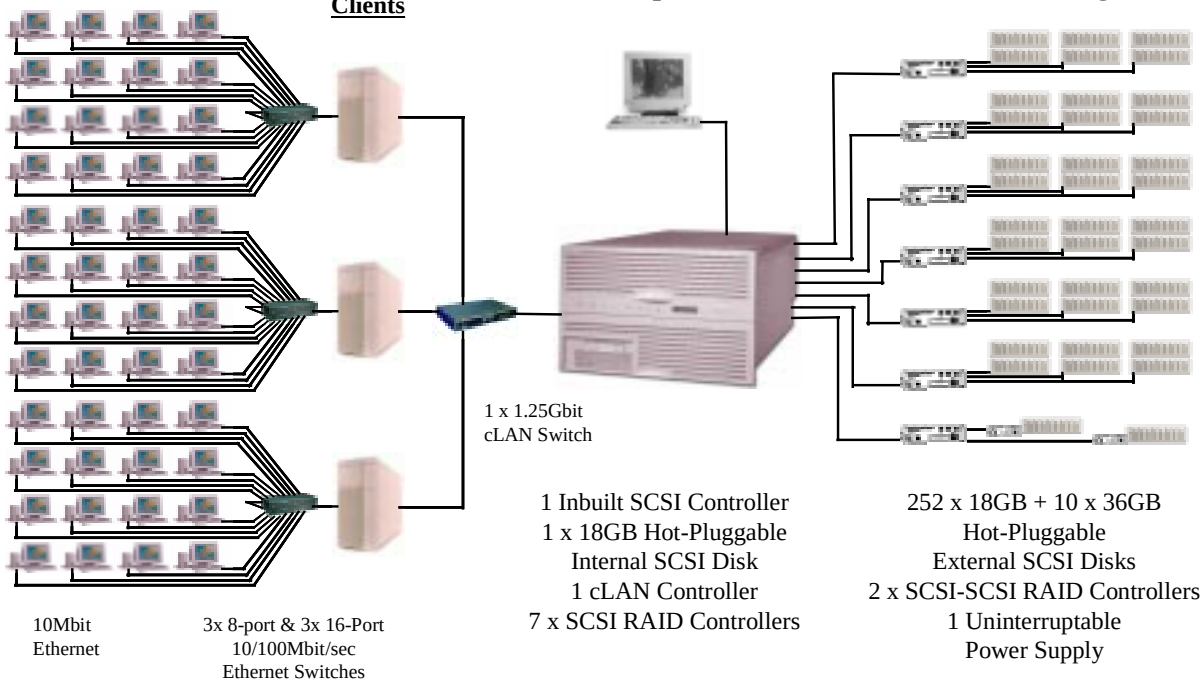
Executive Summary

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 Tom Sawyer of Performance Metrics, Inc. to verify compliance with the relevant TPC specification.

UNISYS	**e-@ction Enterprise Server ES5085R** **(8P 700MHz/2MB)**			TPC-C Rev. 3.5 Report Date: 14-Nov-2000 Revision Date: 18-Jan-2001
Total System Cost	TPC-C Throughput	Price/Performance		Availability Date
$858,408	**61,390.43 tpmC**	**$13.99 per tpmC**		**1-Mar-2001**
Processors	Database Manager	Operating System	Other Software	Number of Users
8 Pentium® III Xeon 700 MHz 2MB L2 cache	Microsoft SQL Server 2000 Enterprise Edition	Microsoft Windows 2000 Datacenter Server	Windows 2000 Server, IIS 5.0 and COM+	**48,960**



System Components	**Enterprise Server**		**Clients** (each of 3)	
	Quantity	Type	Quantity	Type
Processors	8	700 MHz Pentium® III Xeon with 2MB Level 2 Cache	2	866 MHz Pentium® III with 256KB Level 2 Cache
Memory	1	32 GB	1	1024 MB
Disk Controllers	7 + 2 1	SCSI RAID Inbuilt SCSI	1	Inbuilt SCSI
Disk Drives	253 10	17.0 GB 34.0 GB	1	16.9 GB
Total Storage		4640.14 GB		16.9 GB
CD-ROM / Tape	1	SCSI CD-ROM Drive	1	CD-ROM Drive

UNISYS	e-@ction Enterprise Server ES5085R (8P 700MHz/2MB)						TPC-C Rev 3.5 Revision Date: 18-Jan-2001
Description	Style	Third Party Brand	Pricing	Unit Price	Qty.	Extended Price	5 Years Maint.
Server Hardware							
SYS: ES5085R, 0CPU, 0MB	ESR508152-GZN		1	$13,407	1	$13,407	$2,184
PROC: 700MHz Pentium III Xeon /2MB Cache	XEO37002-2MB		1	$3,315	8	$26,520	$9,984
MEM: 1GB Memory, SDRAM, Buf 6ns	DIM6168-1GB		1	$3,094	32	$99,008	$33,024
BRD: Processor Mezzanine Board, 0 Proc.	ESR82-MEZ		1	$958	1	$958	$168
BRD: Memory Carrier Board, 0 Mem.	ESR81-MCB		1	$737	1	$737	$216
MEM: Cache Coherency Filter, 4x SRAM	ESR81-CC4		1	$774	2	$1,548	$528
CTRL: RAID, PCI, 4-Ch w/ 0MB Mem	RAD6004-P64		1	$1,400	7	$9,800	$3,360
MEM: RAID 32MB Cache & Bat. BU	RAD6324-MEB		1	$479	7	$3,353	$1,176
CTRL: cLAN, PCI Host Adapter & Cable	PCI10001-CXG		1	$587	1	$587	$168
DISK: 18GB, 10K SCSI LVD, SCA	HDM18110-CX1		1	$567	1	$567	$192
MAINT: 3 Yr. Performance Gold Svc Warranty Upgrade	WI8003-PGS		1	$1,768	1		$1,768
MONITOR: 15-inch Color	EVG2100-P		1	$221	1	$221	
					Subtotal	**$156,706**	**$52,768**
Storage Hardware							
DISK: 18GB Drive, 15K SCSI LVD, SCA	ESM18308-W45		1	$737	252	$185,724	
DISK: 18GB Drive, 15K SCSI LVD, SCA 10% spares	ESM18308-W45		1	$737	26		$19,162
DISK: 36GB Drive, 10K SCSI LVD, SCA	OSD36209-W45		1	$1,172	10	$11,720	
DISK: 36GB Drive, 10K SCSI LVD, SCA 10% spares	OSD36209-W45		1	$1,172	2		$2,344
CAB: Disk, 8 SCA w/ I/F cards, 0 Disks, 3U	ESM310300-L05		1	$2,100	36	$75,600	$38,448
CAB: Disk, 8 SCA w/ RAID Cntlr, 0MB, 0 Disks, 3U	ESM311000-LR		1	$5,166	2	$10,332	$4,944
MEM: 32MB OSM cache	OSM1032-MEM		1	$171	2	$342	$264
PWR: 2nd Power Supply Upgrade, OSM	OSM3000-BPF		1	$530	2	$1,060	$312
PWR:3000 VA UPS, 3U	UPD30001-SXR		1	$1,897	1	$1,897	$1,260
PWR: Distribution Strip, 220V	SFR9-PWR		1	$111	8	$888	
CBL: SCSI 68-pin VHD Conn's, 5 meter	CBL134-5		1	$165	20	$3,300	
CBL: SCSI 68-pin VHD Conn's, 0.5 meter	CBL134-CAT		1	$73	18	$1,314	
CAB: Rackmount Kit for Disk Cages	OSM3000-RMK		1	$131	38	$4,978	
CAB: 36U x 19" x 34" Cabinet, Open	RM361934-OFE		1	$810	4	$3,240	
DOOR: 36U x 19", Rear	RM3619-RDR		1	$277	4	$1,108	
PNL: 36U x 34" Side Skins, L&R	RM3634-SDS		1	$221	4	$884	
					Subtotal	**$302,387**	**$66,734**
Server Software							
O/S: Microsoft Windows 2000 Datacenter Server	WSD200008-L		1	$22,100	1	$22,100	$24,780
O/S: Win2K Datacenter Annual Subscription	DUS200008-L		1	$6,335	5	$31,675	
SYS MGT: ES7000 Value Add Software	ESS508020-N		1	$368	1	$368	$360
APP: SQL Server 2000, Entrprs Edtn, per Proc. Lcnsng		Microsoft	2	$16,541	8	$132,328	$10,475
					Subtotal	**$186,471**	**$35,615**
Client Hardware							
SYS: ES2024 Tower, w/ 0 Proc., 0MB Mem	ES202141-GZN		1	$1,547	3	$4,641	$1,584
PROC:1x866MHz Pentium III/256KB Cache	CPU3866133-256		1	$368	6	$2,208	$3,312
ACC: Voltage Regulator	VRM3-83		1	$29	3	$87	
MEM: 256 MB SDRAM PC133ECC Memory	DIM13368-256		1	$368	12	$4,416	$2,880
DISK: 18GB SCSI 3.5 Internal	ES2024-18G		1	$553	3	$1,659	$576
CTRL: cLAN, PCI Host Adapter & Cable	PCI10001-CXG		1	$586	3	$1,758	$504
MAINT: 3 Yr. Performance Gold Svc Warranty Upgrade	WI2003-PGS		1	$1,031	3		$3,093
MONITOR:15-inch Color	EVG2100-P		1	$221	3	$663	
					Subtotal	**$15,432**	**$11,949**
Client Software							
O/S: Microsoft Windows 2000 Server		Microsoft	2	$738	3	$2,214	Inc. above
ACC: Microsoft Visual C++ Professional 6.0		Microsoft	2	$549	1	$549	Inc. above
					Subtotal	**$2,763**	**$0**
User Connectivity							
SWITCH: cLAN, 8-port, 1.25Gbit	SWT50082-CXG		1	$5,157	1	$5,157	$1,704
SWITCH: Ethernet, 8-Port 100TX TrueFast + 10% spares	D-Link DSS-8+	Netlux	3	$69	5	$345	spared
SWITCH: Ethernet, 16-Port 10/100TX Fast	ETH410-T16		1	$884	3	$2,652	$1,044
HUB: Ethernet,17-Port 10Base-T + 10% spares	CT1017D1	ArkPC	4	$31	3380	$104,780	spared
					Subtotal	**$112,934**	**$2,748**
					Total	**$776,693**	**$169,814**
Unisys Service Pre-Pay Discount							**($31,815)**
Comark discount			1			**($56,284)**	

Notes:

1. HW Maintenance - Unisys 36 month warranty is upgraded to service level: Standard Performance-Gold. Last 24 months are also at service level: Standard Performance-Gold.
2. All Microsoft maintenance is covered by the maintenance cost of Microsoft SQL Server.
3. 10% or minimum 2 spares are added in place of onsite service (products have a five year return-to-vendor warranty)
4. Pricing: 1 = Comark, 2 = Microsoft, 3 = Netlux, 4 = ArkPC

Five Year Cost of Ownership	**$858,408**
TPC-C Throughput	**61,390.43**
$/tpmC	**$13.99**

The benchmark results and test methodology were audited by Tom Sawyer 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
for
Unisys e-@ction Enterprise Server ES5085R

MQTh, Computed Maximum Qualified Throughput: **61,390.43**
% throughput difference, reported & reproducibility runs: 0.04%

Transaction Mix

New Order	44.86%
Payment	43.05%
Delivery	4.02%
Stock-Level	4.03%
Order-Status	4.00%

Response Times

Transaction	Average	Maximum	90th %ile
New-Order	0.32	4.61	0.61
Payment	0.28	4.64	0.57
Delivery	0.17	1.15	0.31
Stock-Level	0.56	2.35	0.87
Order Status	0.28	3.31	0.57
Menu	0.17	1.15	0.31
Delivery (Deferred)	0.20	1.13	0.31

Response time delay added for emulated components (seconds)

RT Response time	0.1
Menu Response time	0.1

Keying/Think Time Times (seconds)

Transaction	Minimum	Average	Maximum
New-Order	18.00/0	18/12.02	18.04/120.3
Payment	3.00/0	3/12.03	3.03/120.3
Delivery	2.00/0	2/5.05	2.02/50.6
Stock-Level	2.00/0	2/5.07	2.03/50.6
Order-Status	2.00/0	2/10.04	2.02/100.71

Test Duration

Ramp up time	46 minutes
Measurement interval (M)	30 minutes
Transactions (all types) completed during measurement interval	4,104,646
Ramp-down time	114 minutes

Checkpointing:

Number of checkpoints	1
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. New Order Think Time Frequency Distribution Curve 5-4
5.6. Response Time versus Throughput Performance 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. Reproducibility 5-7
5.11. Measurement Interval Duration 5-7
5.12. Regulation of Transaction Mix 5-7
5.13. Transaction Statistics 5-7
5.14. Checkpoint Statistics 5-8

6. Clause 6: SUT, Driver & Communications Definition 6-1
6.1. Remote Terminal Emulator (RTE) Description 6-1
6.2. Emulated Components 6-1
6.3. Functional Diagrams 6-1
6.4. Network Configuration 6-1
6.5. Network Bandwidth 6-1
6.6. 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 0-3
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 Think Time Distribution 5-4
Figure 5.7: Response Time versus Throughput 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: RAID Adapter Disk Configuration 4-5
Table 4.5: Disk Administrator Configuration 4-8
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 5-8

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 ES5085R using Microsoft SQL Server 2000 Enterprise Edition on Microsoft Windows 2000 Datacenter Server.

TPC Benchmark C Overview

The TPC Benchmark™ C Standard Specification Revision 3.5 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. General Items

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.

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.

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 :

- *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,*
- *percentage difference between reported throughput and throughput obtained in reproducibility run,*
- *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.

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.

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:

- *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:

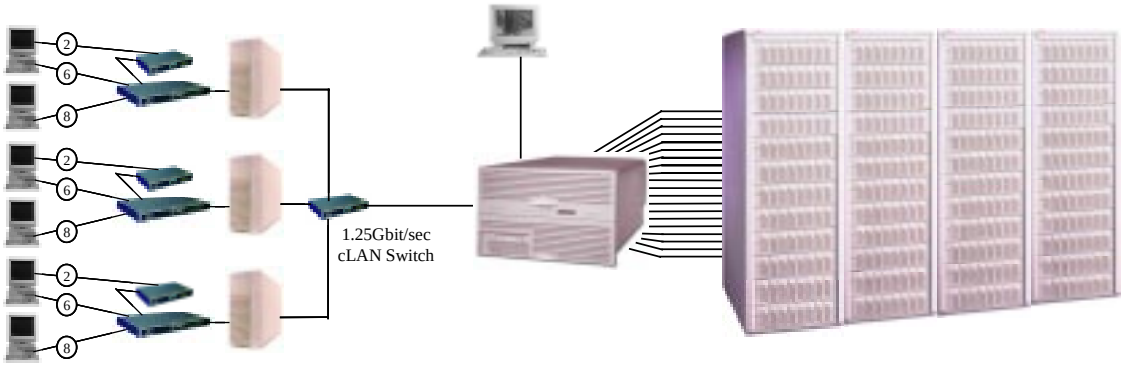
- *Number and type of processors.*
- *Size of allocated memory, and any specific mapping/partitioning of memory unique to the test.*
- *Number and type of disk units (and controllers, if applicable).*
- *Number of channels or bus connections to disk units, including their protocol type.*
- *Number of 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 ES5085R 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 ES5085R is shown in Figure 0.2.

Figure 0.1: Benchmarked Configuration

Unisys e-@ction Enterprise Server ES5085R - Benchmarked Configuration



3x 8-port & 3x 16-Port
10/100Mbit/sec
Ethernet Switches

6 RTEs
emulating
48,960 users
48 LANs w/ 1020
users each

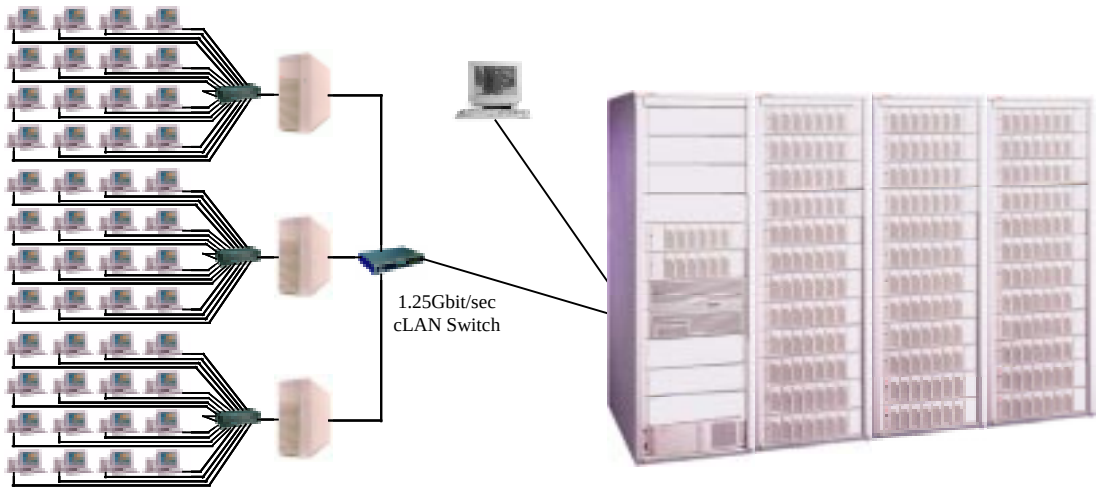
Clients (each of 3)
e-@ction ES2024
2x 866MHz Pentium® III CPU,
256KB L2 cache,
1024MB memory,
1 x 16.9 GB disk,
1 PCI 10/100Mb Ethernet adapter for RTE connections
1 PCI cLAN controller for server connection

Enterprise Server
e-@ction ES5085R
8x Pentium® III Xeon 700MHz CPUs,
2MB L2 cache per CPU,
32 GB memory,
1 Internal SCSI controller,
1 x 17.0 GB internal SCSI disk,
7 PCI SCSI RAID controllers,
1 cLAN controller

Storage
OSM3000
252 x 17.0 GB data,
10 x 34.0 GB log,
30 x 34.0 GB backup,
2 x SCSI-SCSI RAID
Controllers

Figure 0.2: Priced Configuration

Unisys e-@ction Enterprise Server ES5085R - Priced Configuration



48,960 PC's

3x 8-port & 3x 16-Port
10/100Mbit/sec
Ethernet Switches

Clients (each of 3)
e-@ction ES2024
2x 866MHz Pentium® III CPU,
256KB L2 cache,
1024MB memory,
1 x 16.9 GB disk,
1 PCI Fast Ethernet adapter
1 cLAN controller

Enterprise Server
e-@ction ES2085R
8x Pentium® III Xeon 700MHz CPUs,
2MB L2 cache per CPU,
32 GB memory,
1 Internal SCSI controller,
1 x 17.0 GB internal SCSI disk,
7 PCI SCSI RAID controllers,
1 cLAN controller

Storage
OSM3000
252 x 17.0 GB data,
10 x 34.0 GB log,
2 x SCSI-SCSI RAID
Controllers,
1 Uninterruptable Power
Supply

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 data base.

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 data base, must be disclosed.

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 data base 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.

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, any such partitioning must be disclosed.

Partitioning was not used for any table in this implementation.

1.5. Replication, Duplication or Additions

Replication of tables, if used, must be disclosed.

Additional and/or duplicate attributes in any table must be disclosed along with a statement on the impact on performance.

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

2. Clause 2: Transaction & Terminal Profiles

2.1. Random Number Generation

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

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.

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).

This was verified by the auditor.

2.4. Presentation Managers or Intelligent Terminal

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

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 homr and remote order-lines in the New-Order transactions must be disclosed.

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

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

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

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

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.

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

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.

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.

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 system, using a 100 warehouse database with 1000 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 (using a 100 warehouse database on the same system) to demonstrate durability in the case of loss of a log disk and loss of a data disk. The same log disks and controllers were used for the log as for the fully scaled database. Three extra 18GB drives on each of two data controllers of the fully scaled database were used as the data area for the 100 warehouse database.

1. The database was backed up to extra disks 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 1000 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, and Disk Manager was used to assign the proper drive letter to the new volume. 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. Next, samples of the contents of the “success” file from the drivers were compared against corresponding rows of the ORDER table to verify that records in the “success” file for committed New-Order transactions had corresponding records in the ORDER table.
9. 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, must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted (see Clause 4.2.2 and the Auditor's attestation letter) the cardinality of the WAREHOUSE table as initially configured and the number of rows deleted must be disclosed.

The TPC-C database for this test was configured with 4896 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	4,896
District	48,960
Customer	146,880,000
History	146,880,000
Order	146,880,000
New-Order	44,064,000
Order Line	1,468,798,778
Stock	489,600,000
Item	100,000

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.

Tables 4.3, 4.4 and 4.5 list the distribution of the database over 252 18GB disks and the transaction log over 5 mirrored pairs of 36GB disks for the benchmark configuration. In addition, there was one 18GB disk containing Windows 2000 Datacenter Server and SQL Server 2000 code and the Master database plus the paging file. Thirty 18GB disks were present solely for database backup. These components were not used during the measurements and thus were not priced. 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:

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.

No table partitioning or replication was done.

4.6. DBMS Space Requirements

Details of the 180 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).

Appendix E lists the space requirements for the 180-day space as well as the logical log space for eight hours.

Table 4.3: Disk Cage Configuration

Disk Cage Configuration										
Adapter	**Channel**	**Id**	**Id**	**Id**	**Id**	**Id**	**Id**	**Id**	**Id**	**Rack #**
1	**0**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		36GB	36GB	36GB	36GB	36GB	empty	empty	empty	1
	1	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		36GB	36GB	36GB	36GB	36GB	empty	empty	empty	2
	2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		36GB	36GB	36GB	36GB	36GB	36GB	36GB	empty	*
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	*
	3	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		36GB	36GB	36GB	36GB	36GB	36GB	36GB	empty	*
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		36GB	36GB	36GB	36GB	36GB	36GB	36GB	36GB	*
2	**0**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	3
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	4
	1	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	5
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	6
	2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	7
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	8
3	**0**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	9
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	10
	1	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	11
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	12
	2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	13
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	14
4	**0**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	15
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	16
	1	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	17
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	18
	2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	19
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	20

Table 4.3: Disk Cage Configuration (Continued)

Disk Cage Configuration										
Adapter	**Channel**	**Id**	**Id**	**Id**	**Id**	**Id**	**Id**	**Id**	**Id**	**Rack #**
5	**0**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	21
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	22
	1	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	23
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	24
	2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	25
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	26
6	**0**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	27
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	28
	1	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	29
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	30
	2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	31
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	32
7	**0**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	33
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	34
	1	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	35
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	36
	2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	37
		8	**9**	**10**	**11**	**12**	**13**	**14**	**15**	
		18GB	18GB	18GB	18GB	18GB	18GB	18GB	empty	38

Table 4.4: RAID Adapter Disk Configuration

RAID Adapter Disk Configuration							
Adapter	ID	Channel 0	Channel 1	Channel 2	Channel 3	RAID Configuration	Drive Letters
1	0	A1-1	A1-2	A2-1	A3-1	Configure Array 1 as RAID 1 (log)	L:
	1			A2-2	A3-2		
	2			A2-3	A3-3		
	3			A2-4	A3-4		
	4			A2-5	A3-5		
	5			A2-6	A3-6	Configure Array 2 as RAID 5 (backup)	X:
	6			A2-7	A3-7		
	8			A2-8	A3-8		
	9			A2-9	A3-9		
	10			A2-10	A3-10		
	11			A2-11	A3-11	Configure Array 3 as RAID 5 (backup)	Y:
	12			A2-12	A3-12		
	13			A2-13	A3-13		
	14			A2-14	A3-14		
	15			A2-15	A3-15		
2	0	A1-1	A2-1	A3-1		Configure Arrays 1-3 as RAID 0 (data)	E:, K:, R:
	1	A1-2	A2-2	A3-2			
	2	A1-3	A2-3	A3-3			
	3	A1-4	A2-4	A3-4			
	4	A1-5	A2-5	A3-5			
	5	A1-6	A2-6	A3-6			
	6	A1-7	A2-7	A3-7			
	8	A1-8	A2-8	A3-8			
	9	A1-9	A2-9	A3-9			
	10	A1-10	A2-10	A3-10			
	11	A1-11	A2-11	A3-11			
	12	A1-12	A2-12	A3-12			
	13	A1-13	A2-13	A3-13			
	14	A1-14	A2-14	A3-14			
	15						
3	0	A1-1	A2-1	A3-1		Configure Arrays 1-3 as RAID 0 (data)	F:, M:, S:
	1	A1-2	A2-2	A3-2			
	2	A1-3	A2-3	A3-3			
	3	A1-4	A2-4	A3-4			
	4	A1-5	A2-5	A3-5			
	5	A1-6	A2-6	A3-6			
	6	A1-7	A2-7	A3-7			
	8	A1-8	A2-8	A3-8			
	9	A1-9	A2-9	A3-9			
	10	A1-10	A2-10	A3-10			
	11	A1-11	A2-11	A3-11			
	12	A1-12	A2-12	A3-12			
	13	A1-13	A2-13	A3-13			
	14	A1-14	A2-14	A3-14			
	15						

Table 4.4: RAID Adapter Disk Configuration (Continued)

RAID Adapter Disk Configuration							
Adapter	ID	Channel 0	Channel 1	Channel 2	Channel 3	RAID Configuration	Drive Letters
4	0	A1-1	A2-1	A3-1		Configure Arrays 1-3 as RAID 0 (data)	G:, N:, T:
	1	A1-2	A2-2	A3-2			
	2	A1-3	A2-3	A3-3			
	3	A1-4	A2-4	A3-4			
	4	A1-5	A2-5	A3-5			
	5	A1-6	A2-6	A3-6			
	6	A1-7	A2-7	A3-7			
	8	A1-8	A2-8	A3-8			
	9	A1-9	A2-9	A3-9			
	10	A1-10	A2-10	A3-10			
	11	A1-11	A2-11	A3-11			
	12	A1-12	A2-12	A3-12			
	13	A1-13	A2-13	A3-13			
	14	A1-14	A2-14	A3-14			
	15						
5	0	A1-1	A2-1	A3-1		Configure Arrays 1-3 as RAID 0 (data)	H:, O:, U:
	1	A1-2	A2-2	A3-2			
	2	A1-3	A2-3	A3-3			
	3	A1-4	A2-4	A3-4			
	4	A1-5	A2-5	A3-5			
	5	A1-6	A2-6	A3-6			
	6	A1-7	A2-7	A3-7			
	8	A1-8	A2-8	A3-8			
	9	A1-9	A2-9	A3-9			
	10	A1-10	A2-10	A3-10			
	11	A1-11	A2-11	A3-11			
	12	A1-12	A2-12	A3-12			
	13	A1-13	A2-13	A3-13			
	14	A1-14	A2-14	A3-14			
	15						
6	0	A1-1	A2-1	A3-1		Configure Arrays 1-3 as RAID 0 (data)	I:, P:, V:
	1	A1-2	A2-2	A3-2			
	2	A1-3	A2-3	A3-3			
	3	A1-4	A2-4	A3-4			
	4	A1-5	A2-5	A3-5			
	5	A1-6	A2-6	A3-6			
	6	A1-7	A2-7	A3-7			
	8	A1-8	A2-8	A3-8			
	9	A1-9	A2-9	A3-9			
	10	A1-10	A2-10	A3-10			
	11	A1-11	A2-11	A3-11			
	12	A1-12	A2-12	A3-12			
	13	A1-13	A2-13	A3-13			
	14	A1-14	A2-14	A3-14			
	15						

Table 4.4: RAID Adapter Disk Configuration (Continued)

RAID Adapter Disk Configuration							
Adapter	ID	Channel 0	Channel 1	Channel 2	Channel 3	RAID Configuration	Drive Letters
7	0	A1-1	A2-1	A3-1		Configure Arrays 1-3 as RAID 0 (data)	J:, Q:, W:
	1	A1-2	A2-2	A3-2			
	2	A1-3	A2-3	A3-3			
	3	A1-4	A2-4	A3-4			
	4	A1-5	A2-5	A3-5			
	5	A1-6	A2-6	A3-6			
	6	A1-7	A2-7	A3-7			
	8	A1-8	A2-8	A3-8			
	9	A1-9	A2-9	A3-9			
	10	A1-10	A2-10	A3-10			
	11	A1-11	A2-11	A3-11			
	12	A1-12	A2-12	A3-12			
	13	A1-13	A2-13	A3-13			
	14	A1-14	A2-14	A3-14			
	15						

Table 4.5: Disk Management Configuration

Disk Management Configuration				
Disk 0 17.01 GB	C: System NTFS 17.01 GB			
Disk 1 474.79 GB	X: Backup1 NTFS 474.79 GB			
Disk 2 474.79 GB	Y: Backup2 NTFS 474.79 GB			
Disk 3 169.64 GB	L: unknown 117.19 GB			 Unallocated 52.45 GB
Disk 4 713.99 GB	E: unknown 16.90 GB	K: unknown 32.33 GB	R: unknown 20.51 GB	 Unallocated 644.25 GB
Disk 5 713.99 GB	F: unknown 16.90 GB	M: unknown 32.33 GB	S: unknown 20.51 GB	 Unallocated 644.25 GB
Disk 6 713.99 GB	G: unknown 16.90 GB	N: unknown 32.33 GB	T: unknown 20.51 GB	 Unallocated 644.25 GB
Disk 7 713.99 GB	H: unknown 16.90 GB	O: unknown 32.33 GB	U: unknown 20.51 GB	 Unallocated 644.25 GB
Disk 8 713.99 GB	I: unknown 16.90 GB	P: unknown 32.33 GB	V: unknown 20.51 GB	 Unallocated 644.25 GB
Disk 9 713.99 GB	J: unknown 16.90 GB	Q: unknown 32.33 GB	W: unknown 20.51 GB	 Unallocated 644.25 GB
CDRom 0 DVD (D:) Online				

5. Clause 5: Performance Metrics & Response Time

5.1. Measured Throughput (tpmC)

Measured tpmC must be reported.

The measured tpmC was 61,390.43.

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.

Table 5.1: Response Time Data

Transaction	Average	Maximum	90th %ile
New-Order	0.32	4.61	0.61
Payment	0.28	4.64	0.57
Delivery	0.17	1.15	0.31
Stock-Level	0.56	2.35	0.87
Order Status	0.28	3.31	0.57
Menu	0.17	1.15	0.31
Delivery (Deferred)	0.20	1.13	0.31

5.3. Keying and Think Times

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

Table 5.2: Keying Times

Transaction	Minimum	Average	Maximum
New-Order	18.00	18.00	18.04
Payment	3.00	3.00	3.03
Delivery	2.00	2.00	2.02
Stock-Level	2.00	2.00	2.03
Order Status	2.00	2.00	2.02

Table 5.3: Think Times

Transaction	Minimum	Average	Maximum
New-Order	0.00	12.02	120.30
Payment	0.00	12.03	120.30
Delivery	0.00	5.05	50.60
Stock-Level	0.00	5.07	50.60
Order Status	0.00	10.04	100.71

5.4. Response Time Frequency Distribution Curves

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

Figure 5.1: New Order Response Time Distribution

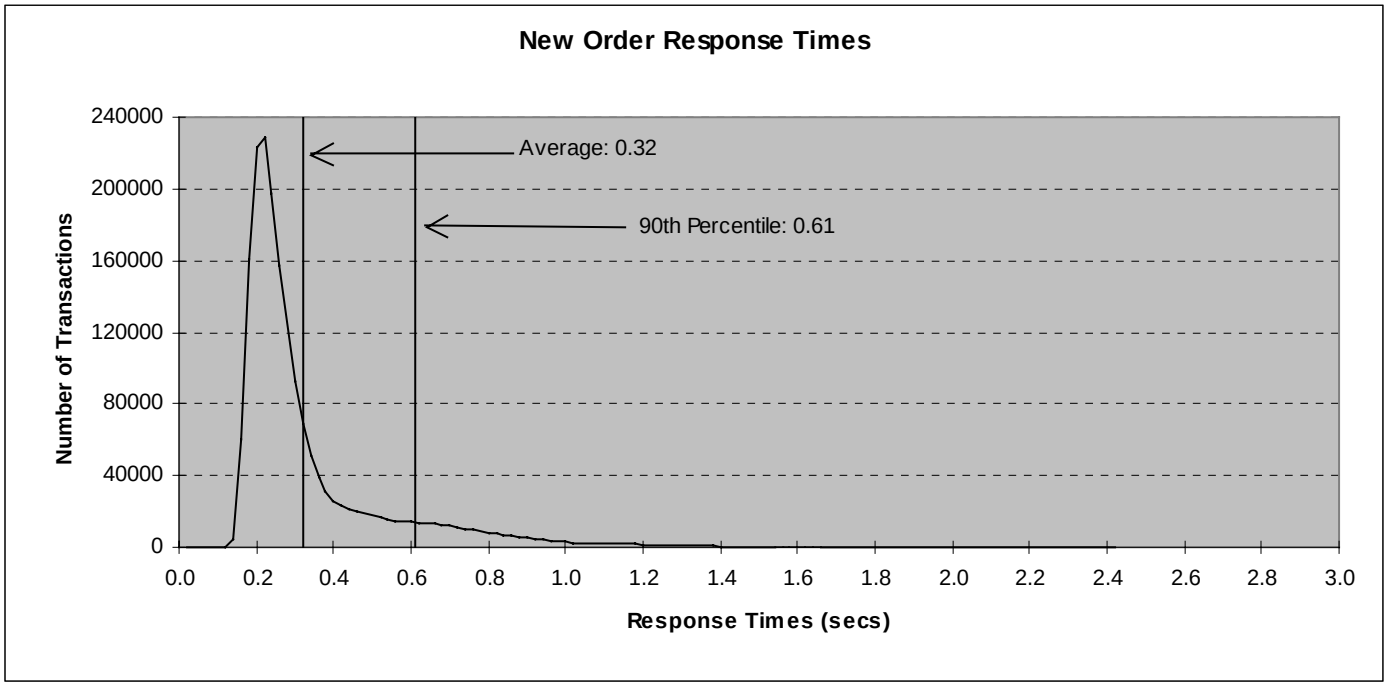


Figure 5.2: Payment Response Time Distribution

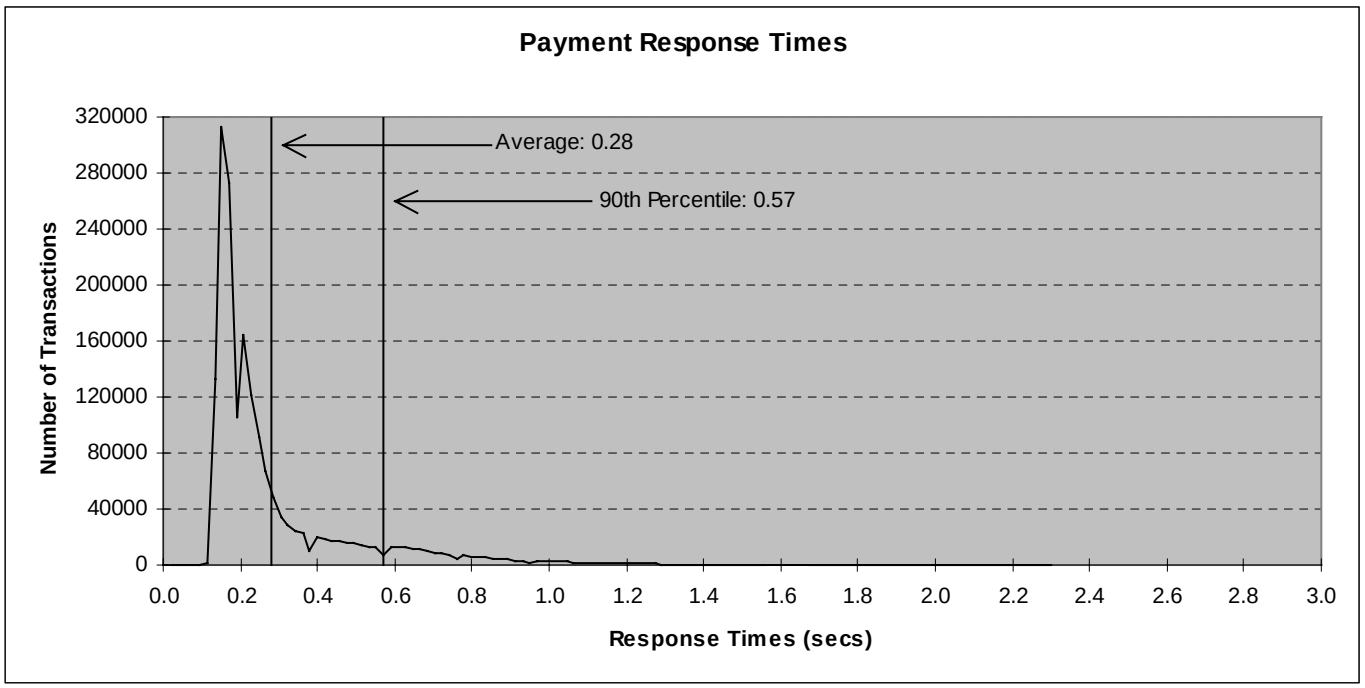


Figure 5.3: Order Status Response Time Distribution

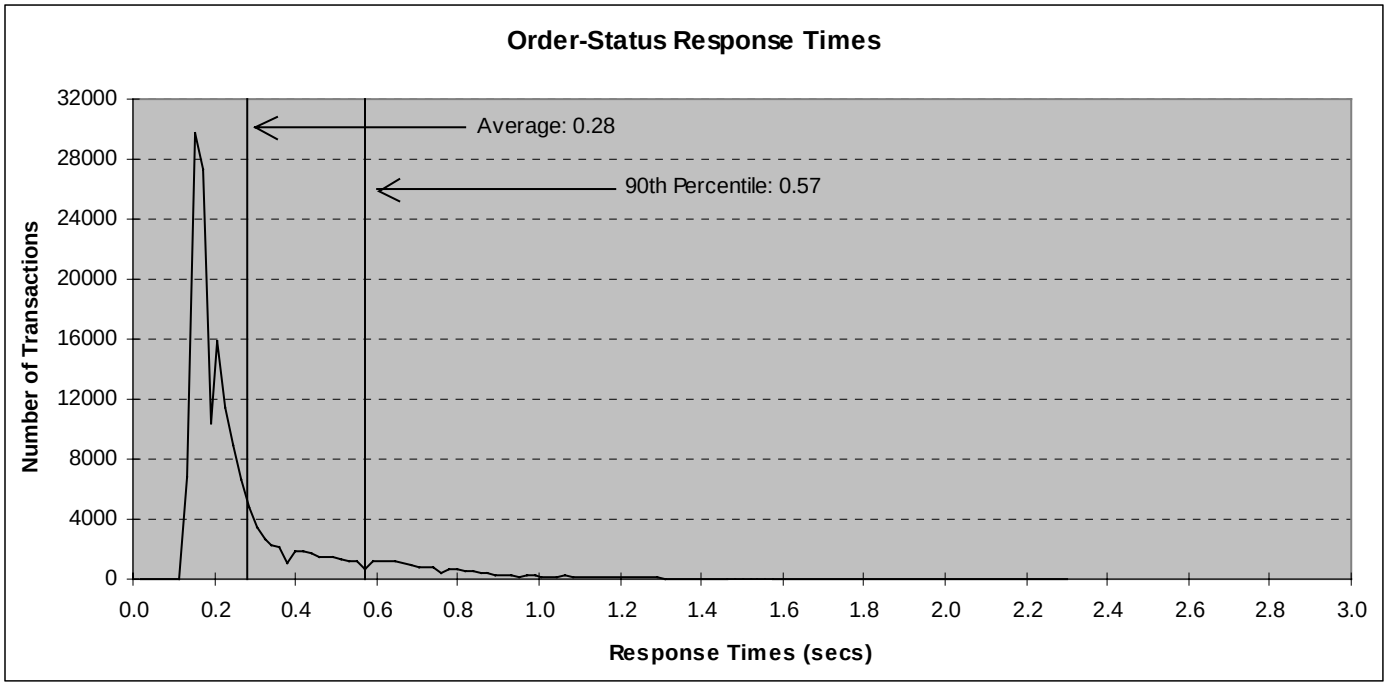


Figure 5.4: Delivery Response Time Distribution

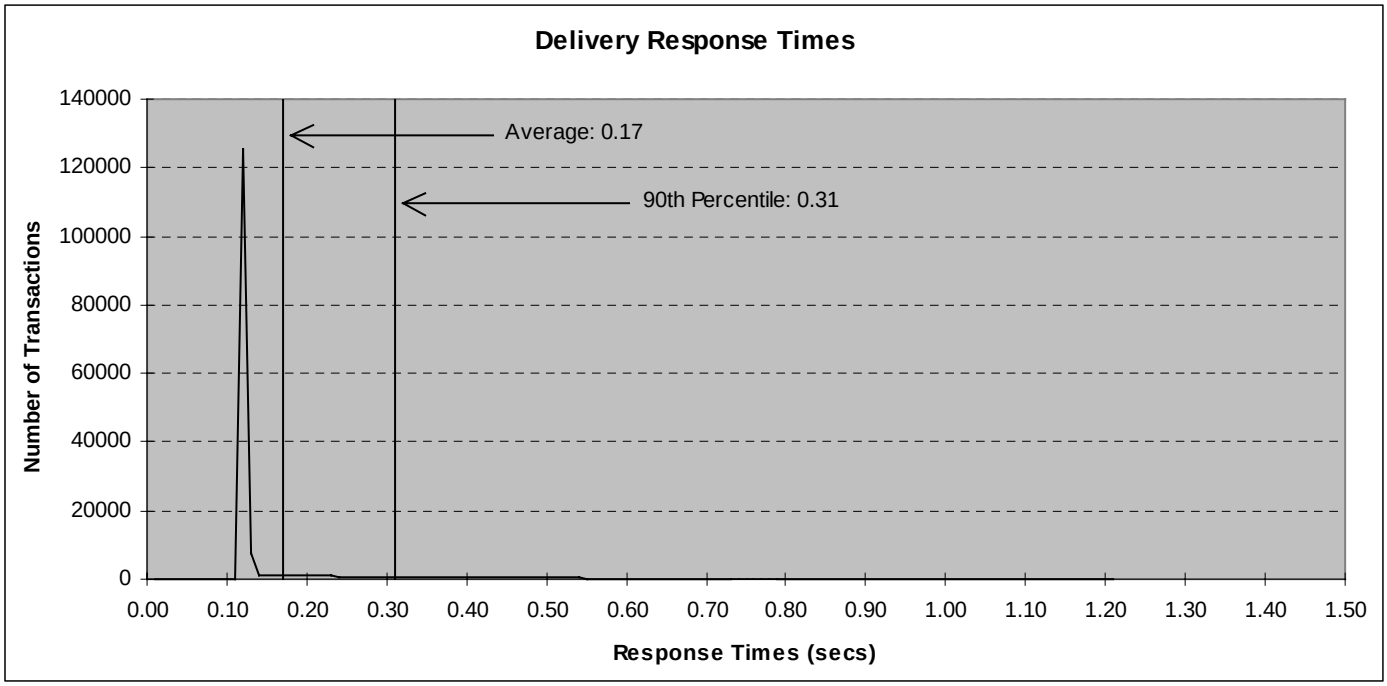
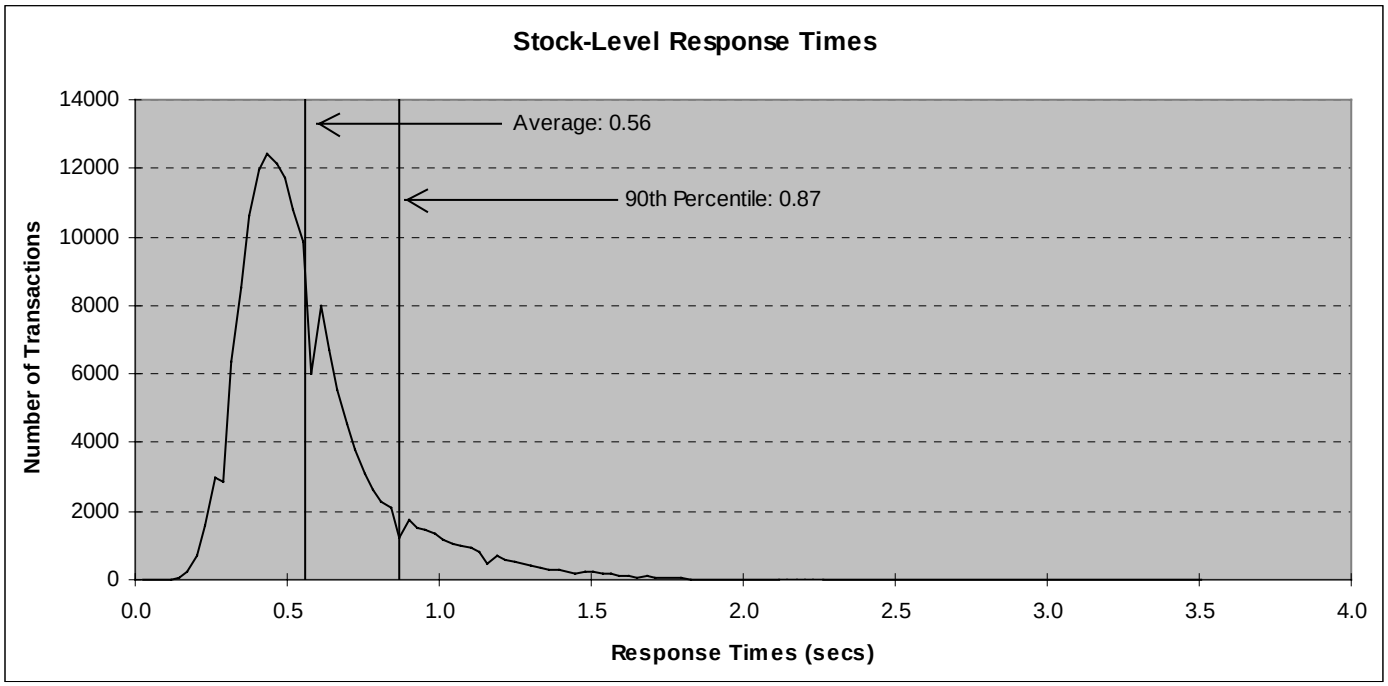


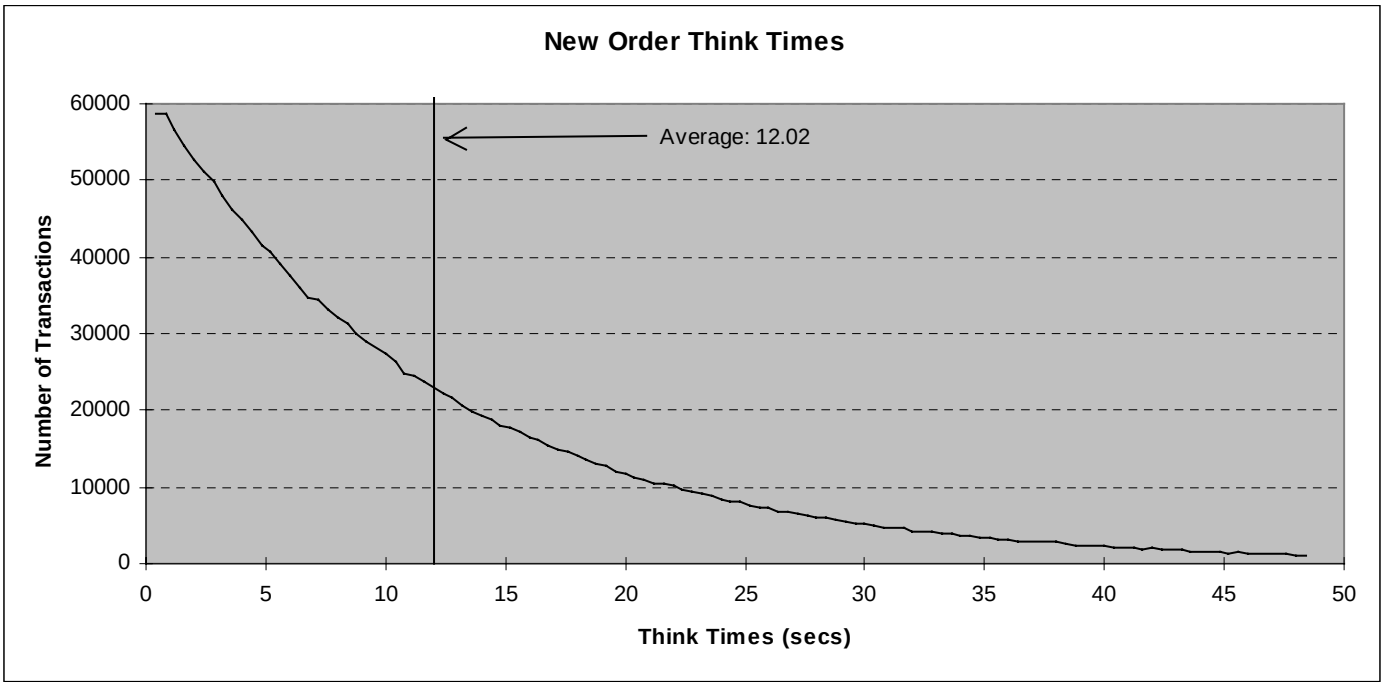
Figure 5.5: Stock Level Response Time Distribution



5.5. 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.

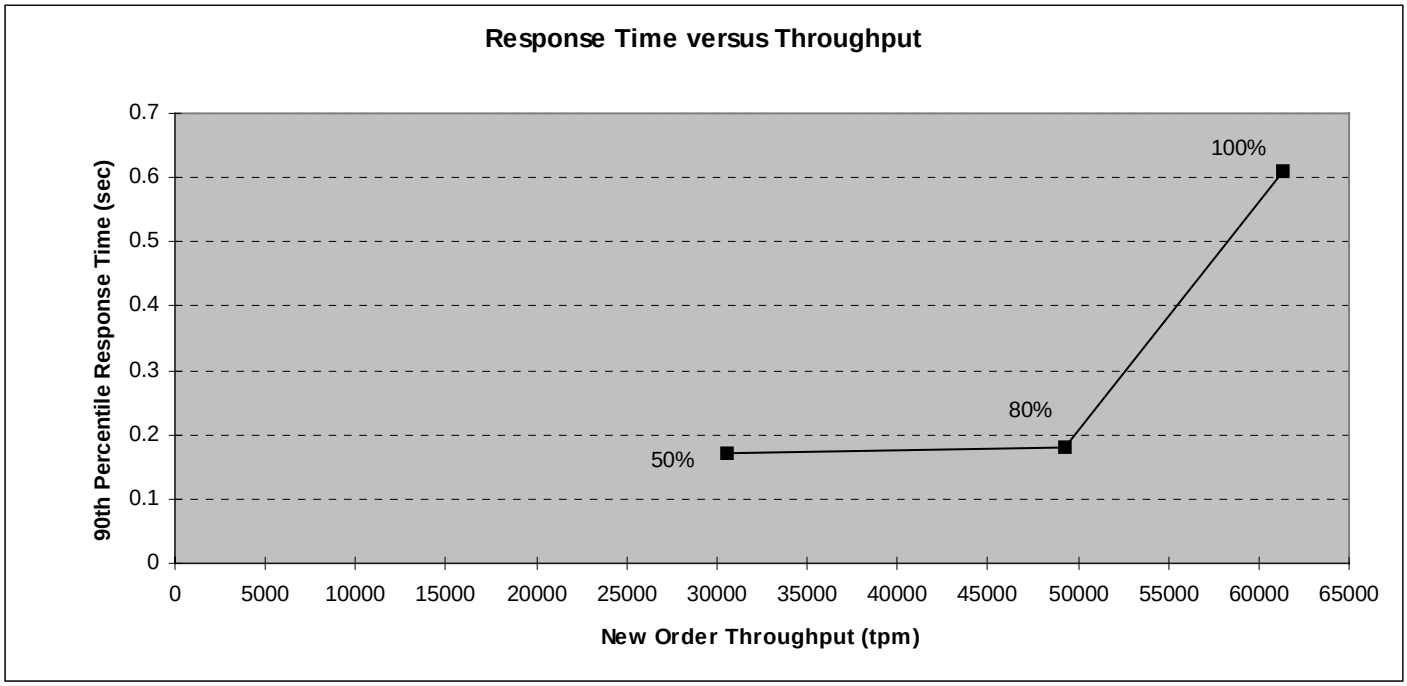
Figure 5.6: New Order Think Time Distribution



5.6. 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

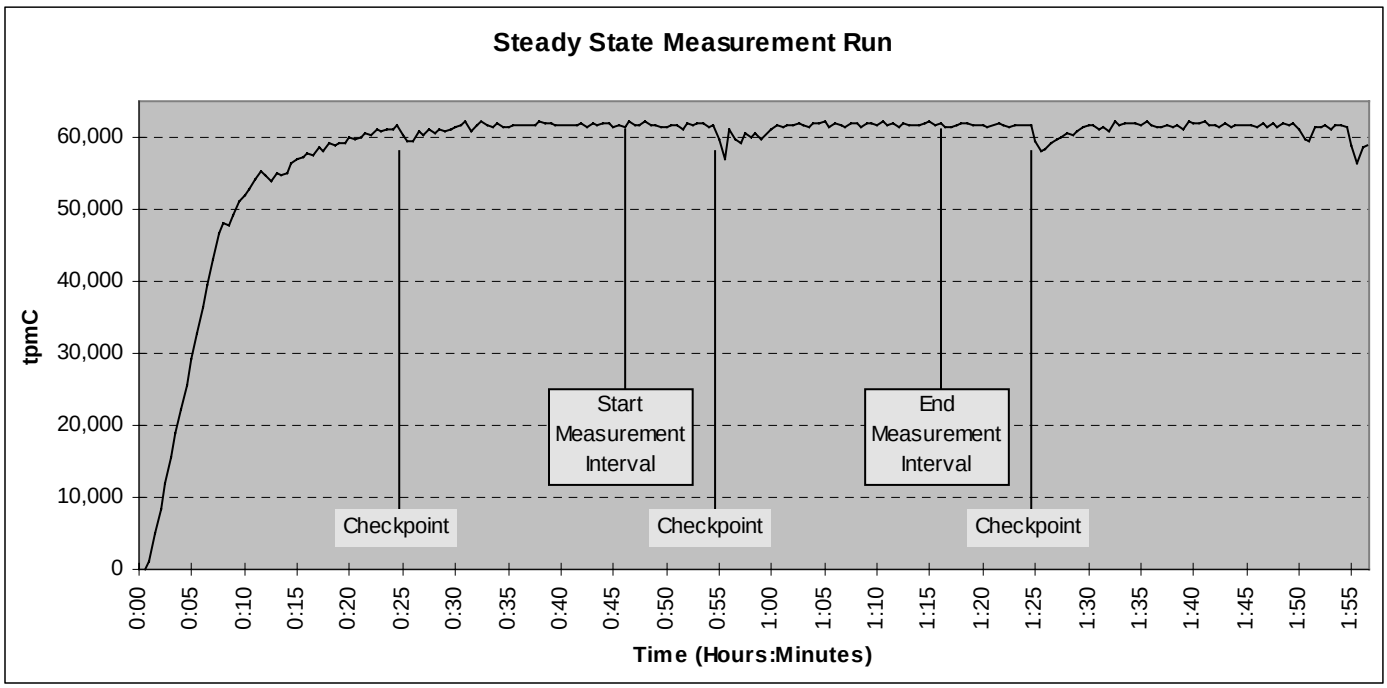
Figure 5.7: Response Time versus Throughput



5.7. New-Order Throughput vs. Time

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

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.

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.

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. Using this information, the positioning of the checkpoint within the measurement interval was verified to be clear of the guard zones.

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. Reproducibility

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

In a repeat test, carried out in the same manner as the primary test, a throughput of 61,363.33 tpmC was achieved on the same database during a 30-minute, steady state run. All required transaction statistics were met. See the Auditor's attestation letter for details.

5.11. Measurement Interval Duration

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

The measurement interval was 30 minutes.

5.12. Regulation of Transaction Mix

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

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

5.13. Transaction Statistics

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

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

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

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

The percentage of remote Payment transactions must be disclosed.

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

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

Table 5.4 shows this information.

Table 5.4: Transaction Statistics

Transaction Type	Statistics	Value
New Order	Rolledback transactions	1.00%
	Home warehouse	99.00%
	Remote warehouse	1.00%
	Average Items per Order	10.00
Payment	Home warehouse	85.03%
	Remote warehouse	14.97%
	Non-primary key access	59.98%
Order Status	Non-primary key access	60.08%
Delivery	Skipped transactions (Interactive)	0
	Skipped transaction counts (Deferred)	0
	Skipped District counts (Deferred)	0
Transaction Mix	New Order	44.86%
	Payment	43.05%
	Delivery	4.02%
	Stock-Level	4.03%
	Order-Status	4.00%

5.14. 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.

There is one checkpoint in the measurement interval. The checkpoint starts 521 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 13.5 minutes. In conformance with Clause 5.5.2.2, the checkpoint occurs outside the guard zones.

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.

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

6.2. 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.

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

6.3. 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.

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

6.4. 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.

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.

6.5. Network Bandwidth

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

Local area networks (LAN) are used in the priced and tested configurations. The database server (SUT) contains a single cLAN 1.25 gigabit per second LAN adapter. This LAN segment runs at 1.25 gigabit per second in both the priced and tested configurations. The clients contain one 10/100 megabit per second LAN adapter and one cLAN 1.25 gigabit per second adapter. The 10/100 LAN connection runs at 100 megabits per second in the priced and tested configurations. The cLAN connection runs at 1.25 gigabit per second in the priced and tested configurations. 48 (16 per client) user LAN segments run at 10 megabits per second in both the priced and tested configurations. One sixteen-port and one eight-port 10/100 megabit per second switch per client were concatenated and used to connect the clients to the users. An additional 1.25 gigabit per second cLAN switch was used to connect the clients to the

database server. In the priced configuration, the clients are connected to workstations (PCs running web browsers). In the tested configuration, the clients are connected to RTE driver systems emulating web browsers.

6.6. Operator Intervention

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

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

7. Clause 7: Pricing

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.

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

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.

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.

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

The five year support pricing for Unisys Corporation Business Server products is based on a 36-month warranty on hardware, upgraded to service level Performance-Gold, plus an additional 24 months of support at service level Performance-Gold. Microsoft support pricing is based on 5 years of annual support costs.

Unisys's standard Performance-Gold service level provides onsite support for hardware from 8:00 A.M. to 5:00 P.M., Monday through Fridays with four hour maximum response for spare parts. Service requests made as late as 5:00 P.M. will receive a response the same day.

Server disks are covered by Comark's 5 year, return-to-factory warranty with seven day maximum replenishment, and appropriate spares are included in the priced configuration. Netlux and MicroBarn also provide 5 year, return-to-factory warranties, seven day replenishment for their respective components, 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 to 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.

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 5-year pricing, price/performance (price/tpmC), and the availability date must be included.

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.

None.

7.5. Usage Pricing

For any usage pricing, the sponsor must disclose:

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

The component pricing based on usage is shown below:

- One (1) Microsoft Windows 2000 Datacenter Server license
- One (1) Microsoft SQL Server 2000 Enterprise Edition license
- Three (3) 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.

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. *Clause 9 : Audit*

9.1. Auditor's Report

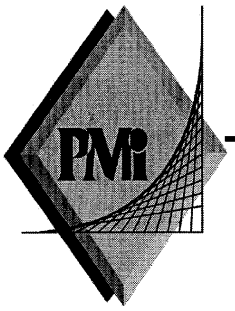
The auditor's name, address, phone number and a copy of the auditor's attestation letter indicating compliance must be included in the Full Disclosure Report.

This implementation of the TPC Benchmark C on the Unisys e-@ction Enterprise Server ES5085R was audited by Tom Sawyer, a TPC certified auditor of:

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

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

The attestation letter is shown on the next 2 pages.



PERFORMANCE METRICS INC.
TPC Certified Auditors

November 1, 2000

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 ES5085R
Database Manager:	Microsoft SQL Server 2000 Enterprise Edition
Operating System:	Microsoft Windows 2000 Datacenter Server
Transaction Manager:	Microsoft COM+ (included in Windows 2000)

Server: e-@ction Enterprise Server ES5085R				
CPU's	Memory	Disks	90% Response	tpmC
8 Pentium III Xeon @ 700 MHz	Main: 32 GB Cache: 2MB each	253 @ 18GB 10 @ 36GB	**0.61 sec**	**61,390.43**
3 Clients: e-@ction ES2024				
2 Pentium III @ 866 MHz	Main: 1 GB Cache: 512K	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 4,896 warehouses.
- The ACID properties were met.

137 Yankton St. Suite 101, Folsom 95630
(916) 985-1131 fax: (916) 985-1185 email: Lorna@PerfMetrics.com

- The durability data loss and log loss tests were performed on a 100-warehouse database.
- 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 180-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 48,960 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 30 minutes.
- One checkpoint was taken before the measured interval.
- One checkpoint was taken during the measured interval.
- The checkpoints were verified to be clear of the guard zone.
- The system pricing was checked for major components and maintenance.

Auditor Notes:

Additional drives were present on the measured system. I verified that they were not used during the measurement..

Sincerely,

Tom Sawyer

Tom Sawyer
Auditor

137 Yankton St. Suite 101, Folsom 95630
(916) 985-1131 fax: (916) 985-1185 email: Lorna@PerfMetrics.com

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

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

```
#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 iTMMaxMsg = 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 * szTMInitError =
   HTTPHdr "<HTML>"
   "<HEAD><TITLE>Welcome To TPC-C</TITLE></HEAD><BODY>"
   "<B>TPCC Extension Error (TMInit 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 iTMMaxSz =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);
         iTMMaxSz = max(iTMMaxSz,sizeof(NEW_ORDER_DATA));
         iTMMaxSz = max(iTMMaxSz,sizeof(PAYMENT_DATA));
         iTMMaxSz = max(iTMMaxSz,sizeof(ORDER_STATUS_DATA));
         iTMMaxSz = max(iTMMaxSz,sizeof(DELIVERY_DATA));
         iTMMaxSz = max(iTMMaxSz,sizeof(STOCK_LEVEL_DATA));
         iTMMaxSz += 10;
         TMonInit(iTMMaxSz);
         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 (TMInit(&pTPCC->tsTMon))
      {
         wsprintf(pDiag,"ThrAtt(%ld): TMInit %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 HttpXit;
      };
   };
   if (pTPCC->tsTMon.pTxnData == NULL)
      SendResponse(pECB,szTMInitError,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=\"TERMID\" VALUE=\"-2\">"
   "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"0\">"
   "Warehouse ID <INPUT NAME=\"w_id\" SIZE=4><BR>"
   "District ID <INPUT NAME=\"d_id\" SIZE=2><BR>"
   "<HR>"
   "<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"Submit\">"
   "</FORM>";

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 * TERMIDTOKEN = "TERMID=";
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->ErrTxt,"");
   return(FALSE);
};   // TPCCClear

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

   pTPCC->iStatusId = STATUS_OK;
   if (GetHidden(pTPCC->RecvMsg,&pTPCC->uFormId,&iSyncId,&iTermId))
   {
      uRslt = TPCCSEND;
      FormatLogin(pTPCC->SendMsg,pTPCC->ErrTxt);
      goto HdlrXit;
   };
   if (iTermId > 0)
   {
      pTerm = TermGet(iTermId);
      if (pTerm == NULL)
      {
         uRslt = TPCCSEND;
         strcpy(pTPCC->ErrTxt,"Invalid Term Id");
         FormatLogin(pTPCC->SendMsg,pTPCC->ErrTxt);
         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->ErrTxt,"Must log in first!");
      FormatLogin(pTPCC->SendMsg,pTPCC->ErrTxt);
      uRslt = TPCCSEND;
      goto HdlrXit;
   };
   // Check for multiple log in attempts
   if (pTPCC->sWId != 0 && uCmdId == CMD_SUBMIT)
   {
      strcpy(pTPCC->ErrTxt,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->ErrTxt,"%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>                                      Payment<BR>"
      "Date: %2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d <BR><BR>"
      "Warehouse: %4.4d"
      "                          District: %2.2d<BR>"
      "%-20s                     %-20s<BR>"
      "%-20s                     %-20s<BR>"
      "%-20s %-2s %10.10s       %-20s %-2s %10.10s<BR><BR>"
      "Customer: %4.4d  Cust-Warehouse: %4.4d  Cust-District: %2.2d<BR>"
      "Name:   %-16s %-2s %-16s     Since:  %2.2d-%2.2d-%4.4d<BR>"
      "        %-20s                     Credit: %-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 += wsprintf(pOut + iInx,
      "<PRE>                                    Delivery<BR>"
      "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)
   {
      wsprintf(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;
      wsprintf(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)
   {
      wsprintf(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 += wsprintf(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 += wsprintf(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>                                Stock-Level<BR>"
      "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 = atol(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.  HTP 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=\"TERMID\" 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>                                  New Order<BR>"
      "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>                                    Payment<BR>"
      "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=\"TERMID\" 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=\"TERMID\" 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

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

#define DEFAULTDQSIZE 2000

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

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 boundry, 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(&pTArray[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];
      wsprintf(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];
      wsprintf(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 TMInit(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 iTMMaxSz;

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


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

//============================================================================
//
// Function name: TMInit
//
//    Result:
//       FALSE    Initialization completed successfully
//       TRUE     Initialization failed
//
//============================================================================
BOOL TMInit(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(iTMMaxSz);
   if (!(pTMon->pTxnData))
   {
      wsprintf(pTMon->pszErrTxt,"COM Allocate TxnData Failed\n");
      return(TRUE);
   };
   pTMon->iSize = iTMMaxSz;
   return(FALSE);
};   // TMInit

//============================================================================
//
// 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_tb1, struct _timeb * p_tb2);
VOID TimebCopy(struct _timeb * p_tbDest, struct _timeb * p_tbSource);
TMILLI TimebElapsed(struct _timeb * p_tb1);
VOID TimebClear(struct _timeb * p_tb1);
CHAR * TimebToString(struct _timeb * p_tb1,CHAR * psz,BOOL bMillis);
BOOL TimebFromString(struct _timeb * p_tb1,CHAR * psz);
VOID TimebAddSecs(struct _timeb * p_tb1,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_tbDest, struct _timeb * p_tbSource)
{
   p_tbDest->time = p_tbSource->time;
   p_tbDest->millitm = p_tbSource->millitm;
   p_tbDest->dstflag = p_tbSource->dstflag;
   p_tbDest->timezone = p_tbSource->timezone;

};   // TimebCopy

//=============================================================================
//
// Function name: TimebDiff
//    Time difference in milliseconds between _timeb _t1 and _timeb _t2.
//
//=============================================================================
TMILLI TimebDiff(struct _timeb * p_tb1, struct _timeb * p_tb2)
{
   LONG lRslt;
   lRslt = ((p_tb2->time - p_tb1->time) * 1000) +
               (p_tb2->millitm - p_tb1->millitm);
   if (lRslt < 0)
      return(0);
   else
      return((TMILLI) lRslt);

};   // TimebDiff

//=============================================================================
//
// Function name: TimebElaspsed
//
//=============================================================================
TMILLI TimebElapsed(struct _timeb * p_tb1)
{
   struct _timeb _tb2;
   _ftime(&_tb2);
   return (TimebDiff(p_tb1,&_tb2));

};   // TimebElapsed

//=============================================================================
//
// Function name: TimebClear
//
//=============================================================================
VOID TimebClear(struct _timeb * p_tb1)
{
   p_tb1->time = 0;
   p_tb1->millitm = 0;

};   // TimebClear

//=============================================================================
//
// Function name: TimebToString
//    Converts timeb to yyyy:mm:dd,hh:mm:ss.sss format
//
//=============================================================================
CHAR * TimebToString(struct _timeb * p_tb1,CHAR * psz,BOOL bMillis)
{
   struct tm * ptm;
   int iInx;
   ptm = localtime(&p_tb1->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_tb1->millitm);
   return(psz);
};   // TimebToString

//=============================================================================
//
// Function name: TimebFromString
//    Converts yyyy:mm:dd,hh:mm:ss.sss (TimebToString) format to timeb
//
//=============================================================================
BOOL TimebFromString(struct _timeb * p_tb1,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_tb1->time = 0;
      p_tb1->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_tb1->millitm = atoi(psz);
   }
   else
      p_tb1->millitm = 0;
   p_tb1->time = mktime(ptm);
   return(FALSE);
};   // TimebFromString

//=========================================================================
//
// Function name: TimebAddSecs
//
//=========================================================================
VOID TimebAddSecs(struct _timeb * p_tb1,INT iSeconds)
{
   p_tb1->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 DEFAULTEVENTLOG 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 ** pTData
            );

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

         HRESULT _stdcall Delivery
            (
               [in] int * iSize,
               [in, out, size_is(,*iSize)] char ** pTData
            );

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

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

         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;
      SQLUINTEGER m_BindOffset;
      SQLUINTEGER 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 ** pTData)
{
   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 commited.");
         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)
         wsprintf(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];
      wsprintf(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)
   {
      wsprintf(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
      )
      {
         wsprintf(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 commited.");
         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_iTPRslt = 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_iTPRslt = 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_iTPRslt = 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_iTPRslt = 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_iTPRslt = 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->bTPRslt = TRUE;
               posd->iTPRslt = 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_iTPRslt = SVCERR_NOORDERS;
      else
      if (m_TData.OSData.c_id == 0 && m_TData.OSData.c_last[0] == 0)
         m_iTPRslt = SVCERR_NOCUSTOMER;
      else
      {
         m_bTPRslt = FALSE;
         m_iTPRslt = SVC_NOERROR;
      };
      break;
   };   // for m_iMaxRetry
   memcpy(posd,&m_TData.OSData,sizeof(ORDER_STATUS_DATA));
   posd->bTPRslt = m_bTPRslt;
   posd->iTPRslt = m_iTPRslt;
   if (!m_bTPRslt)
   {
      strcpy(posd->execution_status,"Transaction commited.");
      return(S_OK);
   };
   if (m_iTPRslt == 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)
         wsprintf(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];
      wsprintf(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 )
   {
      wsprintf(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
      )
   {
      wsprintf(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)
   {
      wsprintf(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];
            wsprintf(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];
            wsprintf(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 commited.");
         return(S_OK);
      };
      if (m_iTPRslt == SVCERR_DEADLOCK)
         wsprintf(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->bTPRslt = TRUE;
      psld->iTPRslt = 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)

backup database tpcc to tpcc2back1, tpcc2back2, tpcc2back3, tpcc2back4
                    with init, stats = 1

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','tpcc2back1','X:\tpcc2_backup1.dmp'
go
exec sp_addumpdevice 'disk','tpcc2back2','Y:\tpcc2_backup2.dmp'
go
exec sp_addumpdevice 'disk','tpcc2back3','X:\tpcc2_backup3.dmp'
go
exec sp_addumpdevice 'disk','tpcc2back4','Y:\tpcc2_backup4.dmp'
go
```

CREATEDB.SQL

```
--  File:     CREATEDB.SQL
--            Microsoft TPC-C Benchmark Kit Ver. 4.21
--            Copyright Microsoft, 1999, 2000
--  Purpose:  Creates tpcc database and backup files
--             for 4896 warehouses.


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:\MSSQL70_tpcc_root.mdf",
        SIZE            = 10MB,         FILEGROWTH      = 0),
FILEGROUP       MSSQL70_misc_fg
(       NAME            = MSSQL70_misc1,
        FILENAME        = "E:",
        SIZE            = 17300MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_misc2,
        FILENAME        = "F:",
        SIZE            = 17300MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_misc3,
        FILENAME        = "G:",
        SIZE            = 17300MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_misc4,
        FILENAME        = "H:",
        SIZE            = 17300MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_misc5,
```

```
        FILENAME        = "I:",
        SIZE            = 17300MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_misc6,
        FILENAME        = "J:",
        SIZE            = 17300MB,      FILEGROWTH      = 0),
FILEGROUP       MSSQL70_cs_fg
(       NAME            = MSSQL70_cs1,
        FILENAME        = "O:",
        SIZE            = 33100MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_cs2,
        FILENAME        = "P:",
        SIZE            = 33100MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_cs3,
        FILENAME        = "Q:",
        SIZE            = 33100MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_cs4,
        FILENAME        = "K:",
        SIZE            = 33100MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_cs5,
        FILENAME        = "M:",
        SIZE            = 33100MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_cs6,
        FILENAME        = "N:",
        SIZE            = 33100MB,      FILEGROWTH      = 0),
FILEGROUP       MSSQL70_ord_fg
(       NAME            = MSSQL70_ord1,
        FILENAME        = "W:",
        SIZE            = 21000MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_ord2,
        FILENAME        = "V:",
        SIZE            = 21000MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_ord3,
        FILENAME        = "U:",
        SIZE            = 21000MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_ord4,
        FILENAME        = "T:",
        SIZE            = 21000MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_ord5,
        FILENAME        = "S:",
        SIZE            = 21000MB,      FILEGROWTH      = 0),
(       NAME            = MSSQL70_ord6,
        FILENAME        = "R:",
        SIZE            = 21000MB,      FILEGROWTH      = 0)
LOG ON
(       NAME            = MSSQL70_tpcc_log,
        FILENAME        = "L:",
        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'
        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 'tpcc2back1', delfile
exec sp_dropdevice 'tpcc2back2', delfile
exec sp_dropdevice 'tpcc2back3', delfile
exec sp_dropdevice 'tpcc2back4', delfile
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)

restore database tpcc from tpcc2back1, tpcc2back2, tpcc2back3, tpcc2back4
                    with replace, stats = 1

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

go
```

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'=='builddb' goto builddb
@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

:builddb
@@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, builddb, 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 runnning 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  rows
from    sysindexes
where   id      = object_id("warehouse")
go

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

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

print   'ITEM TABLE = 100,000'
```

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

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

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

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

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

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

select  rows
from    sysindexes
where   id      =object_id("history")
go

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

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

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

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

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

select  rows
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   history
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_c1' )
       drop index customer.customer_c1

create unique clustered index customer_c1 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_nc1' )
       drop index customer.customer_nc1

create unique nonclustered index customer_nc1 on customer(c_w_id, c_d_id,
c_last, c_first, c_id)
       on 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_c1' )
       drop index district.district_c1

create unique clustered index  district_c1 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_c1' )
       drop index history.history_c1

create unique clustered index history_c1 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_c1' )
       drop index item.item_c1

create unique clustered index item_c1 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_c1' )
       drop index new_order.new_order_c1

create unique clustered index new_order_c1 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_c_id, o_w_id, o_d_id, o_id)
       with pad_index, fillfactor=90
       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
```

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_c1' )
       drop index stock.stock_c1

create unique clustered index stock_c1 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_c1' )
       drop index warehouse.warehouse_c1

create unique clustered index warehouse_c1 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 updlock)
                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

--  accummulate 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 availible 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   = s_quantity -
@li_qty +
                                                         case when (s_quantity -
@li_qty < 10) then 91 else 0 end,
                        s_order_cnt    = s_order_cnt + 1,
                        s_remote_cnt   = s_remote_cnt + case when
(@li_s_w_id = @w_id) then 0 else 1 end,
                        @s_data        = s_data,
                        @s_dist        = case @d_id
                                           when 1  then s_dist_01
                                           when 2  then s_dist_02
                                           when 3  then s_dist_03
                                           when 4  then s_dist_04
                                           when 5  then s_dist_05
                                           when 6  then s_dist_06
                                           when 7  then s_dist_07
                                           when 8  then s_dist_08
                                           when 9  then s_dist_09
                                           when 10 then s_dist_10
                                           end
                where   s_i_id         = @li_id and
                        s_w_id         = @li_s_w_id

--  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 (repeatableread)
        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,
                              @threshhold    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      <  @threshhold

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                    =  DEFLDPACKSIZE;
        pargs->starting_warehouse   =  DEF_STARTING_WAREHOUSE;
        pargs->build_index                  =  BUILD_INDEX;
        pargs->index_order                  =  INDEX_ORDER;
        pargs->index_script_path    =  INDEX_SCRIPT_PATH;
        pargs->scale_down                   =  SCALE_DOWN;

    /* check for zero command line args */
    if ( argc == 1 )
                GetArgsLoaderUsage();

    for (i = 1; i < argc; ++i)
    {
                if (argv[i][0] != '-' && argv[i][0] != '/')
                {
                printf("\nUnrecognized command");
                GetArgsLoaderUsage();
                exit(1);
                }

                ptr = argv[i];

                switch (ptr[1])
                {
                case 'h':       /* Fall 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
%ld\n", (long) BATCH);
        printf("-p TDS packet size
%ld\n", (long) DEFLDPACKSIZE);
        printf("-f Loader Results Output Filename
%s\n", LOADER_RES_FILE);
        printf("-s Starting Warehouse
%ld\n", (long) DEF_STARTING_WAREHOUSE);
        printf("-i Build Option (data = 0, data and index = 1)
%ld\n", (long) BUILD_INDEX);
        printf("-o Cluster Index Build Order (before = 1, after = 0)
%ld\n", (long) INDEX_ORDER);
        printf("-c Build Scaled Database (normal = 0, tiny = 1)
%ld\n", (long) SCALE_DOWN);
        printf("-d Index Script Path
%s\n", INDEX_SCRIPT_PATH);
        printf("-t Table to Load                                        all
tables \n");
    printf("   [item|warehouse|customer|orders]\n");
    printf("   Notes: \n");
    printf("     - the '-t' parameter may be included multiple times to
\n");
    printf("       specify multiple tables to be loaded \n");
    printf("     - 'item' loads ITEM table \n");
    printf("     - 'warehouse' loads WAREHOUSE, DISTRICT, and STOCK tables
\n");
    printf("     - 'customer' loads CUSTOMER and HISTORY tables \n");
    printf("     - 'orders' load NEW-ORDER, ORDERS, ORDER-LINE tables
\n");

        printf("\nNote:  Command line switches are case sensitive.\n");

    exit(0);
}
```

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

//Orginal 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 prercentage 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                              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   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);

        rc = bcp_bind(i_hdbc1, (BYTE *) i_data, 0, I_DATA_LEN, NULL, 0, 0,
5);
        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 != SUCCEED)
                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 != SUCCEED)
                        HandleErrorDBC(w_hdbc1);
        }

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

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

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

        rc = bcp_bind(w_hdbc1, (BYTE *) d_street_1, 0, ADDRESS_LEN, NULL,
0, 0, 4);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) d_street_2, 0, ADDRESS_LEN, NULL,
0, 0, 5);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

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

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

        rc = bcp_bind(w_hdbc1, (BYTE *) d_zip, 0, ZIP_LEN, NULL, 0, 0, 8);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);
```

```
        rc = bcp_bind(w_hdbc1, (BYTE *) &d_tax, 0, SQL_VARLEN_DATA, NULL,
0, SQLFLT8, 9);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_ytd, 0, SQL_VARLEN_DATA, NULL,
0, SQLFLT8, 10);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_next_o_id, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT4, 11);
        if (rc != SUCCEED)
                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);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_05, 0, S_DIST_LEN, NULL, 0,
0, 8);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_06, 0, S_DIST_LEN, NULL, 0,
0, 9);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_07, 0, S_DIST_LEN, NULL, 0,
0, 10);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_08, 0, S_DIST_LEN, NULL, 0,
0, 11);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_09, 0, S_DIST_LEN, NULL, 0,
0, 12);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_10, 0, S_DIST_LEN, NULL, 0,
0, 13);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_ytd, 0, SQL_VARLEN_DATA, NULL,
0, SQLINT4, 14);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_order_cnt, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 15);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) &s_remote_cnt, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 16);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        rc = bcp_bind(w_hdbc1, (BYTE *) s_data, 0, S_DATA_LEN, NULL, 0, 0,
17);
        if (rc != SUCCEED)
                HandleErrorDBC(w_hdbc1);

        s_ytd = s_order_cnt = s_remote_cnt = 0;

        time_start = (TimeNow() / MILLI);

        printf("...Loading stock table\n");

        for (s_i_id=1; s_i_id <= max_items; s_i_id++)
        {

                for (s_w_id = (short)aptr->starting_warehouse; s_w_id <=
aptr->num_warehouses; s_w_id++)
                {

                        s_quantity = (short)RandomNumber(10L,100L);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_01);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_02);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_03);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_04);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_05);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_06);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_07);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_08);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_09);
                        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_10);

                        len = MakeOriginalAlphaString(26,50, S_DATA_LEN,
s_data,10);

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

                        stock_rows_loaded++;
                        CheckForCommit(w_hdbc1, w_hstmt1,
stock_rows_loaded, "stock", &time_start);

                }
        }

        rcint = bcp_done(w_hdbc1);
        if (rcint < 0)
                HandleErrorDBC(w_hdbc1);

        printf("Finished loading stock table.\n");

        SQLFreeStmt(w_hstmt1, SQL_DROP);
```

```
        SQLDisconnect(w_hdbc1);
        SQLFreeConnect(w_hdbc1);

        // if build index after load...
        if ((aptr->build_index == 1) && (aptr->index_order == 0))
                BuildIndex("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 != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_middle, 0, MIDDLE_NAME_LEN,NULL, 0,
0, 5);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_last, 0, LAST_NAME_LEN, NULL, 0, 0,
6);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_street_1, 0, ADDRESS_LEN, NULL, 0,
0, 7);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_street_2, 0, ADDRESS_LEN, NULL,0,0,
8);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_city, 0, ADDRESS_LEN, NULL, 0, 0,
9);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_state, 0, STATE_LEN, NULL, 0, 0,
10);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_zip, 0, ZIP_LEN, NULL, 0, 0, 11);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_phone, 0, PHONE_LEN, NULL, 0, 0,
12);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_since, 0, C_SINCE_LEN, NULL, 0,
SQLCHARACTER, 13);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_credit, 0, CREDIT_LEN, NULL, 0, 0,
14);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_credit_lim, 0, SQL_VARLEN_DATA,
NULL, 0, SQLFLT8, 15);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_discount, 0, SQL_VARLEN_DATA, NULL,
0, SQLFLT8, 16);
        if (rc != SUCCEED)
                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 != SUCCEED)
        //      HandleErrorDBC(c_hdbc1);
    rc = bcp_bind(c_hdbc1, (BYTE *) c_balance, 0, 5, NULL, 0,
SQLCHARACTER, 17);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);


    rc = bcp_bind(c_hdbc1, (BYTE *) &c_ytd_payment, 0, SQL_VARLEN_DATA,
NULL, 0, SQLFLT8, 18);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_payment_cnt, 0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 19);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_delivery_cnt,0, SQL_VARLEN_DATA,
NULL, 0, SQLINT2, 20);
        if (rc != SUCCEED)
                HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_data, 0, 500, NULL, 0, 0, 21);
        if (rc != SUCCEED)
                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,

(LPTHREAD_START_ROUTINE) LoadOrdersTable,

&orders_time_start,
                                                                          0,

&dwThreadID[0]);

                     if (hThread[0] == NULL)
                     {
                            printf("Error, failed in creating creating
thread = 0.\n");
                            exit(-1);
                     }
```

```
                        // start NewOrder table thread

                        printf("...Loading New-Order Table for: d_id = %d,
w_id = %d\n", d_id, w_id);

                        hThread[1] = CreateThread(NULL,
                                                                        0,
(LPTHREAD_START_ROUTINE) LoadNewOrderTable,
&new_order_time_start,
                                                                        0,
&dwThreadID[1]);

                        if (hThread[1] == NULL)
                        {
                                printf("Error, failed in creating creating
thread = 1.\n");
                                exit(-1);
                        }

                        // start Order-Line table thread

                        printf("...Loading Order-Line Table for: d_id = %d,
w_id = %d\n", d_id, w_id);

                        hThread[2] = CreateThread(NULL,
                                                                        0,
(LPTHREAD_START_ROUTINE) LoadOrderLineTable,
&order_line_time_start,
                                                                        0,
&dwThreadID[2]);

                        if (hThread[2] == NULL)
                        {
                                printf("Error, failed in creating creating
thread = 2.\n");
                                exit(-1);
                        }

                        WaitForSingleObject( hThread[0], INFINITE );
                        WaitForSingleObject( hThread[1], INFINITE );
                        WaitForSingleObject( hThread[2], INFINITE );

                        if (CloseHandle(hThread[0]) == FALSE)
                        {
                                printf("Error, failed in closing Orders
thread handle with errno: %d\n", GetLastError());
                        }

                        if (CloseHandle(hThread[1]) == FALSE)
                        {
                                printf("Error, failed in closing NewOrder
thread handle with errno: %d\n", GetLastError());
                        }

                        if (CloseHandle(hThread[2]) == FALSE)
                        {
                                printf("Error, failed in closing OrderLine
thread handle with errno: %d\n", GetLastError());
                        }

                }
        }

        printf("Finished loading orders.\n");


    return;
}


//============================================================================
//
// Function   : OrdersBufInit
//
// Clears shared buffer for ORDERS, NEWORDER, and ORDERLINE
//
//============================================================================

void OrdersBufInit()
{
        int     i;
        int             j;

        for (i=0;i<orders_per_district;i++)
        {
                orders_buf[i].o_id = 0;
                orders_buf[i].o_d_id = 0;
                orders_buf[i].o_w_id = 0;
                orders_buf[i].o_c_id = 0;
                orders_buf[i].o_carrier_id = 0;
                orders_buf[i].o_ol_cnt = 0;
                orders_buf[i].o_all_local = 0;

                for (j=0;j<=14;j++)
                {
                        orders_buf[i].o_ol[j].ol = 0;
                        orders_buf[i].o_ol[j].ol_i_id = 0;
                        orders_buf[i].o_ol[j].ol_supply_w_id = 0;
                        orders_buf[i].o_ol[j].ol_quantity = 0;
                        orders_buf[i].o_ol[j].ol_amount = 0;
                        strcpy(orders_buf[i].o_ol[j].ol_dist_info,"");
                }
        }


}


//============================================================================
//
// Function   : OrdersBufLoad
//
// Fills shared buffer for ORDERS, NEWORDER, and ORDERLINE
//
//============================================================================
```

```
void OrdersBufLoad(int d_id, int w_id)
{

        int     cust[ORDERS_PER_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
(%.2f rps)\n",
                        aptr->batch,
                        table_name,
                                time_diff,
                        rows_loaded,
                                (float) aptr->batch / (time_diff ? time_diff
: 1L));

                *time_start = time_end;
    }

    return;
}


//=============================================================================
//
// Function   : OpenConnections
//
//=============================================================================

void OpenConnections()
{

        RETCODE         rc;

        char                    szDriverString[300];
        char                    szDriverStringOut[1024];
        SQLSMALLINT             cbDriverStringOut;


        SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE, &henv );

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

        SQLAllocHandle(SQL_HANDLE_DBC, henv , &i_hdbc1);
        SQLAllocHandle(SQL_HANDLE_DBC, henv , &w_hdbc1);
        SQLAllocHandle(SQL_HANDLE_DBC, henv , &c_hdbc1);
        SQLAllocHandle(SQL_HANDLE_DBC, henv , &c_hdbc2);
        SQLAllocHandle(SQL_HANDLE_DBC, henv , &o_hdbc1);
        SQLAllocHandle(SQL_HANDLE_DBC, henv , &o_hdbc2);
        SQLAllocHandle(SQL_HANDLE_DBC, henv , &o_hdbc3);

        SQLSetConnectAttr(i_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );
        SQLSetConnectAttr(w_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );
        SQLSetConnectAttr(c_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );
        SQLSetConnectAttr(c_hdbc2, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );
        SQLSetConnectAttr(o_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );
```

```
        SQLSetConnectAttr(o_hdbc2, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );
        SQLSetConnectAttr(o_hdbc3, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON,
SQL_IS_INTEGER );

        // Open connections to SQL Server

        // Connection 1

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

        rc = SQLSetConnectOption (i_hdbc1, SQL_PACKET_SIZE, aptr-
>pack_size);
        if (rc != SUCCEED)
                HandleErrorDBC(i_hdbc1);

        rc = SQLDriverConnect ( i_hdbc1,
                                        NULL,
                                            (SQLCHAR*)&szDriverString[0]
,
                                            SQL_NTS,

        (SQLCHAR*)&szDriverStringOut[0],
                                            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_UINTEGER );
        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:\MSSQL\BINN\SQLSERVR.EXE -c -x -T3502 –g70

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
- -g 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.

SQL Server Stack Size

The default stack size for Microsoft SQL Server 2000 was changed using the EDITBIN utility. The EDITBIN utility ships with Microsoft Visual C++ V6.0. The command used to change the stack size is:

editbin /S: 131072 sqlservr.exe

This command is fully documented as an article in the Microsoft Knowledge Base on the Microsoft Web Site at www.Microsoft.com/support.

BOOT.INI

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

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

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

 ------------------------------
 Oct 30 2000 11:07:10:483AM

(1 row affected)

1> 2> 3>
select @@version



 --------------------------------------------------------------------------
-----
       --------------------------------------------------------------------
-----
       --------------------------------------------------------------------
-----
       ---------------------------------
 Microsoft SQL Server  2000 - 8.00.194 (Intel X86)
```

```
        Aug  6 2000 00:57:48
        Cop
        yright (c) 1988-2000 Microsoft Corporation
        Enterprise Edition on Windo
        ws NT 5.0 (Build 2195: Service Pack 1)



(1 row affected)
1> 2>
1> 2> 3> 4> 5> 6> 7> 8> 9> 10>
--  File:     CONFIG.SQL
--            Microsoft TPC-C Benchmark Kit Ver. 4.21
--            Copyright Microsoft, 1999, 2000
--  Purpose:  Collects SQL Server configuration parameters


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

 ------------------------------
 Oct 30 2000 11:07:11:467AM

(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.

sp_configure "show advanced",1
1> 2> reconfigure with override
1> 2> sp_configure
 name
minimum     maximum     config_value run_value
 ------------------------------------------------------------------------ --
--------- ----------- ------------ -----------
 affinity mask
0  2147483647          255         255
 allow updates
0           1            1           1
 awe enabled
0           1            1           1
 c2 audit mode
0           1            0           0
 cost threshold for parallelism
0       32767            5           5
 cursor threshold
-1  2147483647           -1          -1
 default full-text language
0  2147483647         1033        1033
 default language
0        9999           0           0
 fill factor (%)
0         100           0           0
 index create memory (KB)
704  2147483647            0           0
 lightweight pooling
0           1            1           1
 locks
5000  2147483647            0           0
 max degree of parallelism
0          32            1           1
 max server memory (MB)
4  2147483647        31500       31500
 max text repl size (B)
0  2147483647        65536       65536
 max worker threads
32       32767          184         184
 media retention
0         365            0           0
 min memory per query (KB)
512  2147483647          512         512
 min server memory (MB)
0  2147483647         1500        1500
 nested triggers
0           1            0           0
 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           20          20
 recovery interval (min)
0       32767           54          54
 remote access
0           1            0           0
 remote login timeout (s)
0  2147483647           20          20
 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>
```

External RAID Configuration Parameters

```
<Heading>      External RAID Configuration Parameters


*********************************************************************
*        Unisys Ultra-Wide RAID Controller   OSM1200-RAD            *
*********************************************************************

CPU type:      5x86-133 (WB)

Firmware version 2.22V
Bootcode version 1.12H

Total cache:   32 MB

- Cache        Write Back:           enabled
               optimization:         sequential (128K stripe size)

- Raid         Rebuild Priority:     low
               Write Priority        on Initialization:    disabled
                                     on Rebuild:           disabled
                                     on Normal:            disabled

Logical Volume Partition table

       Volume ID1     Capacity 173733 MB     RAID 0         # drives: 5

Host LUN Assignment

       SCSI Chl       LUN     LVIDx   PortIdx   Capacity
           0           0       1          0     173733 MB

Physical Drives

       Slot   Chl   Id    Capacity   Status   XferRate    Vendor/Product
Id              Firmware
              0     8    34746 MB    online   83.3 MB     UNISYS
036434MAF3364MC    0602
              0     9    34746 MB    online   83.3 MB     UNISYS
036434MAF3364MC    0602
              0    10    34746 MB    online   83.3 MB     UNISYS
036434MAF3364MC    0602
              0    11    34746 MB    online   83.3 MB     UNISYS
036434MAF3364MC    0602
              0    12    34746 MB    online   83.3 MB     UNISYS
036434MAF3364MC    0602
```

Configuration of Log Drives

A single RAD6004-P64 (Mylex ExtremeRAID 2000) controller was used in the SUT for the mirrored log drives. Half of the drives were in one disk cage connected to one channel of the controller and half were in a second disk cage connected to a second channel of the controller. The controller implemented the RAID 1 mirroring across the two channels. Write caching was disabled on both the controller and on all the physical drives themselves.

Two OSM311000-LR disk cages, which include one OSM1200-RAD SCSI-to-SCSI RAID controller each, were used for the log. Each of these controllers implemented RAID 0 striping on the five 36GB drives that were in each disk cage, so that the ExtremeRAID 2000 controller in the SUT saw just two large disks. Each of the OSM1200-RAD controllers had a 32MB cache. Configuration options were set for Write Back caching and Optimized for Sequential IO. The OSM1200-RAD controllers used an algorithm that ensured that cached write data was held for no more than a fraction of a minute before being written to the physical drives.

For the priced configuration, each of the disk cages contained two redundant power supplies. Only one was required to be functional to keep the OSM1200-RAD controller and disk drives operational. A UPS was priced to provide power to one power supply in each disk cage. The second power supply in each disk cage was connected to normal wall power. Thus neither interruption of power nor failure of the UPS would affect the two log disk cages (or their controllers and disks). Since the two disk cages were completely independent of each other, this configuration ensured that there was no single point of failure in writing to the log.

Windows 2000 Datacenter Server Configuration Information

```
System Information report written at: 10/30/2000 02:48:57 PM
[System Information]

[ Following are sub-categories of this main category ]

[System Summary]

Item    Value
OS Name Microsoft Windows 2000 Datacenter Server
Version 5.0.2195 Service Pack 1 Build 2195
OS Manufacturer        Microsoft Corporation
System Name    CAPRICORN8
System Manufacturer    Intel
System Model   0CPRF100 MP SERVER
System Type    X86-based PC
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
```

```
Processor      x86 Family 6 Model 10 Stepping 1 GenuineIntel ~700 Mhz
BIOS Version   OCPRF100- PhoenixBIOS 4.0 Release 6.0
Windows Directory      C:\WINNT
System Directory       C:\WINNT\System32
Boot Device    \Device\Harddisk0\Partition1
Locale  United States
User Name      CAPRICORN8\Administrator
Time Zone      Pacific Standard Time
Total Physical Memory  33,291,580 KB
Available Physical Memory     298,008 KB
Total Virtual Memory   68,304,452 KB
Available Virtual Memory      2,649,420 KB
Page File Space        35,012,872 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
4       Direct memory access controller       OK
2       Standard floppy disk 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
0x0000-0x0CF7  PCI bus OK
0x0D00-0x4000  PCI bus OK
0xA000-0xFFFF  PCI bus OK
0x2000-0x2FFF  DEC 21152 PCI to PCI bridge   OK
0x2000-0x2FFF  Intel(R) PRO/100+ Dual Port Server Adapter   OK
0x3000-0x3FFF  DEC 21154 PCI to PCI bridge   OK
0x3000-0x3FFF  Mylex eXtremeRAID 2000 Disk Array Controller OK
0x1000-0x10FF  Symbios Logic 896, 22910 PCI SCSI Adapter    OK
0x1400-0x14FF  Symbios Logic 896, 22910 PCI SCSI Adapter    OK
0x03B0-0x03BB  Cirrus Logic 5446 Compatible Graphics Adapter       OK
0x03C0-0x03DF  Cirrus Logic 5446 Compatible Graphics Adapter       OK
0x0A79-0x0A79  ISAPNP Read Data Port  OK
0x0279-0x0279  ISAPNP Read Data Port  OK
0x0274-0x0277  ISAPNP Read Data Port  OK
0x00B3-0x00B3  Motherboard resources  OK
0x0C10-0x0C3F  Motherboard resources  OK
0x0CA8-0x0CAF  Motherboard resources  OK
0x0CC0-0x0CCF  Motherboard resources  OK
0x0010-0x001F  Direct memory access controller       OK
0x0080-0x009F  Direct memory access controller       OK
0x00C0-0x00DF  Direct memory access controller       OK
0x0070-0x0077  System CMOS/real time clock   OK
0x0020-0x0021  Programmable interrupt controller     OK
0x0024-0x0025  Programmable interrupt controller     OK
0x0028-0x0029  Programmable interrupt controller     OK
0x002C-0x002D  Programmable interrupt controller     OK
0x0030-0x0031  Programmable interrupt controller     OK
0x0034-0x0035  Programmable interrupt controller     OK
0x0038-0x0039  Programmable interrupt controller     OK
0x003C-0x003D  Programmable interrupt controller     OK
0x00A0-0x00A1  Programmable interrupt controller     OK
0x00A4-0x00A5  Programmable interrupt controller     OK
0x00A8-0x00A9  Programmable interrupt controller     OK
0x00AC-0x00AD  Programmable interrupt controller     OK
0x00B0-0x00B1  Programmable interrupt controller     OK
0x00B4-0x00B5  Programmable interrupt controller     OK
0x00B8-0x00B9  Programmable interrupt controller     OK
0x00BC-0x00BD  Programmable interrupt controller     OK
0x04D0-0x04D1  Programmable interrupt controller     OK
0x00F0-0x00FF  Numeric data processor OK
0x0040-0x0043  System timer   OK
0x0050-0x0053  System timer   OK
0x0061-0x0061  System speaker 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
0x03F2-0x03F5  Standard floppy disk controller       OK
0x03F7-0x03F7  Standard floppy disk controller       OK
0x03F8-0x03FF  Communications Port (COM1)    OK
0x0378-0x037B  Printer Port (LPT1)    OK
0x02F8-0x02FF  Communications Port (COM2)    OK
0x1820-0x182F  Intel(r) 82371AB/EB PCI Bus Master IDE Controller    OK
0x01F0-0x01F7  Primary IDE Channel    OK
0x03F6-0x03F6  Primary IDE Channel    OK
0x1800-0x181F  Intel 82371AB/EB PCI to USB Universal Host Controller
       OK
0x4000-0x5FFF  PCI bus OK
0x4000-0x5FFF  DEC 21154 PCI to PCI bridge   OK
0x4000-0x5FFF  Mylex eXtremeRAID 2000 Disk Array Controller OK
0x5000-0x5FFF  DEC 21154 PCI to PCI bridge   OK
0x5000-0x5FFF  Mylex eXtremeRAID 2000 Disk Array Controller OK
0x6000-0x7FFF  PCI bus OK
0x6000-0x7FFF  DEC 21154 PCI to PCI bridge   OK
0x6000-0x7FFF  Mylex eXtremeRAID 2000 Disk Array Controller OK
0x7000-0x7FFF  DEC 21154 PCI to PCI bridge   OK
0x7000-0x7FFF  Mylex eXtremeRAID 2000 Disk Array Controller OK
0x8000-0x9FFF  PCI bus OK
0x8000-0x9FFF  DEC 21154 PCI to PCI bridge   OK
0x8000-0x9FFF  Mylex eXtremeRAID 2000 Disk Array Controller OK
0x9000-0x9FFF  DEC 21154 PCI to PCI bridge   OK
0x9000-0x9FFF  Mylex eXtremeRAID 2000 Disk Array Controller OK

[IRQs]

IRQ Number     Device
9      Microsoft ACPI-Compliant System
61     Intel(R) PRO/100+ Dual Port Server Adapter
```

```
54      Mylex eXtremeRAID 2000 Disk Array Controller
58      Symbios Logic 896, 22910 PCI SCSI Adapter
18      Symbios Logic 896, 22910 PCI SCSI Adapter
8       System CMOS/real time clock
13      Numeric data processor
1       Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
12      Microsoft PS/2 Mouse
6       Standard floppy disk controller
4       Communications Port (COM1)
3       Communications Port (COM2)
14      Primary IDE Channel
49      Intel 82371AB/EB PCI to USB Universal Host Controller
50      cLAN Host Adapter
40      Mylex eXtremeRAID 2000 Disk Array Controller
36      Mylex eXtremeRAID 2000 Disk Array Controller
32      Mylex eXtremeRAID 2000 Disk Array Controller
28      Mylex eXtremeRAID 2000 Disk Array Controller
24      Mylex eXtremeRAID 2000 Disk Array Controller
20      Mylex eXtremeRAID 2000 Disk Array Controller

[Memory]

Range   Device  Status
0xA0000-0xBFFFF         PCI bus OK
0xA0000-0xBFFFF         Cirrus Logic 5446 Compatible Graphics Adapter
        OK
0xC8000-0xDFFFF         PCI bus OK
0xE0000-0xFFFFF         PCI bus OK
0xF0000000-0xF4FFFFFF   PCI bus OK
0xF0000000-0xF4FFFFFF   Symbios Logic 896, 22910 PCI SCSI Adapter    OK
0xFFF00000-0xFFFFFFFF   PCI bus OK
0xF0100000-0xF02FFFFF   DEC 21152 PCI to PCI bridge   OK
0xF0100000-0xF02FFFFF   Intel(R) PRO/100+ Dual Port Server Adapter   OK
0xF4000000-0xF40FFFFF   DEC 21152 PCI to PCI bridge   OK
0xF4000000-0xF40FFFFF   Intel(R) PRO/100+ Dual Port Server Adapter   OK
0xF0800000-0xF0FFFFFF   DEC 21154 PCI to PCI bridge   OK
0xF0800000-0xF0FFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xF4800000-0xF4FFFFFF   DEC 21154 PCI to PCI bridge   OK
0xF4800000-0xF4FFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xF0005400-0xF00057FF   Symbios Logic 896, 22910 PCI SCSI Adapter    OK
0xF0005800-0xF0005BFF   Symbios Logic 896, 22910 PCI SCSI Adapter    OK
0xF0002000-0xF0003FFF   Symbios Logic 896, 22910 PCI SCSI Adapter    OK
0xF2000000-0xF3FFFFFF   Cirrus Logic 5446 Compatible Graphics Adapter
        OK
0xF0004000-0xF0004FFF   Cirrus Logic 5446 Compatible Graphics Adapter
        OK
0xF5000000-0xF8FFFFFF   PCI bus OK
0xF5000000-0xF8FFFFFF   cLAN Host Adapter      OK
0xF5020000-0xF503FFFF   cLAN Host Adapter      OK
0xF5200000-0xF53FFFFF   cLAN Host Adapter      OK
0xF6000000-0xF6FFFFFF   cLAN Host Adapter      OK
0xF7000000-0xF77FFFFF   DEC 21154 PCI to PCI bridge   OK
0xF7000000-0xF77FFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xF8000000-0xF87FFFFF   DEC 21154 PCI to PCI bridge   OK
0xF8000000-0xF87FFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xF7800000-0xF7FFFFFF   DEC 21154 PCI to PCI bridge   OK
0xF7800000-0xF7FFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xF8800000-0xF8FFFFFF   DEC 21154 PCI to PCI bridge   OK
0xF8800000-0xF8FFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xF9000000-0xFB7FFFFF   PCI bus OK
0xF9800000-0xF9FFFFFF   DEC 21154 PCI to PCI bridge   OK
0xF9800000-0xF9FFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xFA800000-0xFAFFFFFF   DEC 21154 PCI to PCI bridge   OK
0xFA800000-0xFAFFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xFA000000-0xFA7FFFFF   DEC 21154 PCI to PCI bridge   OK
0xFA000000-0xFA7FFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xFB000000-0xFB7FFFFF   DEC 21154 PCI to PCI bridge   OK
0xFB000000-0xFB7FFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xFB800000-0xFDFFFFFF   PCI bus OK
0xFC000000-0xFC7FFFFF   DEC 21154 PCI to PCI bridge   OK
0xFC000000-0xFC7FFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xFD000000-0xFD7FFFFF   DEC 21154 PCI to PCI bridge   OK
0xFD000000-0xFD7FFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xFC800000-0xFCFFFFFF   DEC 21154 PCI to PCI bridge   OK
0xFC800000-0xFCFFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller OK
0xFD800000-0xFDFFFFFF   DEC 21154 PCI to PCI bridge   OK
0xFD800000-0xFDFFFFFF   Mylex eXtremeRAID 2000 Disk Array Controller 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)  10/3/2000 3:38:25 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)  8/8/2000 5:00:00 AM
c:\winnt\system32\lhacm.acm   Microsoft Corporation         OK
        C:\WINNT\System32\LHACM.ACM   4.4.3385       33.27 KB (34,064
bytes)  10/3/2000 3:38:26 PM
c:\winnt\system32\tssoft32.acm        DSP GROUP, INC.               OK
        C:\WINNT\System32\TSSOFT32.ACM        1.01    9.27 KB (9,488 bytes)
        8/8/2000 5:00:00 AM
c:\winnt\system32\msgsm32.acm Microsoft Corporation         OK
        C:\WINNT\System32\MSGSM32.ACM 5.00.2134.1    22.27 KB (22,800
bytes)  8/8/2000 5:00:00 AM
c:\winnt\system32\msg711.acm  Microsoft Corporation         OK
        C:\WINNT\System32\MSG711.ACM  5.00.2134.1    10.27 KB (10,512
bytes)  8/8/2000 5:00:00 AM
c:\winnt\system32\msadp32.acm Microsoft Corporation         OK
        C:\WINNT\System32\MSADP32.ACM 5.00.2134.1    14.77 KB (15,120
bytes)  8/8/2000 5:00:00 AM
c:\winnt\system32\imaadp32.acm        Microsoft Corporation         OK
        C:\WINNT\System32\IMAADP32.ACM        5.00.2134.1    16.27 KB
(16,656 bytes) 8/8/2000 5:00:00 AM

[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)        8/8/2000 5:00:00 AM
c:\winnt\system32\msh261.drv  Microsoft Corporation         OK
        C:\WINNT\System32\MSH261.DRV  4.4.3385       163.77 KB (167,696
bytes)  10/3/2000 3:38:25 PM
c:\winnt\system32\msh263.drv  Microsoft Corporation         OK
        C:\WINNT\System32\MSH263.DRV  4.4.3385       252.27 KB (258,320
bytes)  10/3/2000 3:37:54 PM
c:\winnt\system32\msvidc32.dll       Microsoft Corporation          OK
        C:\WINNT\System32\MSVIDC32.DLL       5.00.2134.1    27.27 KB
(27,920 bytes) 8/8/2000 5:00:00 AM
c:\winnt\system32\msrle32.dll Microsoft Corporation         OK
        C:\WINNT\System32\MSRLE32.DLL 5.00.2134.1    10.77 KB (11,024
bytes)  8/8/2000 5:00:00 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)  8/8/2000 5:00:00 AM
c:\winnt\system32\iccvid.dll  Radius Inc.            OK
        C:\WINNT\System32\ICCVID.DLL  1.10.0.6       108.00 KB (110,592
bytes)  8/8/2000 5:00:00 AM

[CD-ROM]

Item    Value
Drive   D:
Description    CD-ROM Drive
Media Loaded   False
Media Type     CD-ROM
Name    TOSHIBA DVD-ROM SD-M1402
Manufacturer   (Standard CD-ROM drives)
Status  OK
Transfer Rate  Not Available
SCSI Target ID 0
PNP Device ID  IDE\CDROMTOSHIBA_DVD-ROM_SD-
M1402_____________1010____\30363030303037393739202020202020202020202020


[Sound Device]

Item    Value
No sound devices

[Display]

Item    Value
Name    Cirrus Logic 5446 Compatible Graphics Adapter
PNP Device ID
        PCI\VEN_1013&DEV_00B8&SUBSYS_00B81013&REV_45\3&267A616A&0&60
Adapter Type   Cirrus Logic 5446BE, Cirrus Logic compatible
Adapter Description    Cirrus Logic 5446 Compatible Graphics Adapter
Adapter RAM    2.00 MB (2,097,152 bytes)
Installed Drivers      vga.sys,cirrus.sys,vga256.dll,vga64k.dll
Driver Version 5.00.2146.1
INF File       display.inf (cirrus section)
Color Planes   1
Color Table Entries    65536
Resolution     1024 x 768 x 60 hertz
Bits/Pixel     16


[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&43B47AD&0
NumberOfFunctionKeys   12


[Pointing Device]

Item    Value
Hardware Type  Microsoft PS/2 Mouse
Number of Buttons      2
Status  OK
PNP Device ID  ACPI\PNP0F03\4&43B47AD&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] RAS Async Adapter
Adapter Type   Not Available
Product Name   RAS Async Adapter
Installed      True
PNP Device ID  Not Available
Last Reset     10/29/2000 11:20:16 PM
Index   0
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   [00000001] WAN Miniport (L2TP)
Adapter Type   Not Available
Product Name   WAN Miniport (L2TP)
Installed      True
PNP Device ID  ROOT\MS_L2TPMINIPORT\0000
Last Reset     10/29/2000 11:20:16 PM
Index  1
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 (50320, 5.00.2179.1)

Name   [00000002] 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     10/29/2000 11:20:16 PM
Index  2
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 (47376, 5.00.2160.1)

Name   [00000003] Direct Parallel
Adapter Type   Not Available
Product Name   Direct Parallel
Installed      True
PNP Device ID  ROOT\MS_PTIMINIPORT\0000
Last Reset     10/29/2000 11:20:16 PM
Index  3
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   [00000004] WAN Miniport (IP)
Adapter Type   Not Available
Product Name   WAN Miniport (IP)
Installed      True
PNP Device ID  ROOT\MS_NDISWANIP\0000
Last Reset     10/29/2000 11:20:16 PM
Index  4
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 (89808, 5.00.2184.1)

Name   [00000005] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type   Ethernet 802.3
Product Name   Intel(R) PRO/100+ Dual Port Server Adapter
Installed      True
PNP Device ID
       PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&27BD01E1&0&2020
Last Reset     10/29/2000 11:20:16 PM
Index  5
Service Name   E100B
IP Address     192.168.91.161
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    00:90:27:D3:0C:19
Service Name   E100B
IRQ Number     61
I/O Port       0x2000-0x2FFF
Driver  c:\winnt\system32\drivers\e100bnt5.sys (85776, 4.02.38.0000)

Name   [00000006] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type   Not Available
Product Name   Intel(R) PRO/100+ Dual Port Server Adapter
Installed      True
PNP Device ID
       PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&27BD01E1&0&2820
Last Reset     10/29/2000 11:20:16 PM
Index  6
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    Not Available
Service Name   E100B
Driver  c:\winnt\system32\drivers\e100bnt5.sys (85776, 4.02.38.0000)

Name   [00000007] 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     10/29/2000 11:20:16 PM
Index  7
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 (89808, 5.00.2184.1)

Name   [00000008] 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     10/29/2000 11:20:16 PM
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 (89808, 5.00.2184.1)

Name   [00000009] 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     10/29/2000 11:20:16 PM
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 (89808, 5.00.2184.1)

Name   [00000010] 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&20
Last Reset     10/29/2000 11:20:16 PM
Index  10
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:00:08:86
Service Name   GNINDIS
IRQ Number     50
Driver  c:\winnt\system32\drivers\gnindis.sys (21616, 4.01.00)

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\0001
Last Reset     10/29/2000 11:20:16 PM
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 (89808, 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\0002
Last Reset     10/29/2000 11:20:16 PM
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 (89808, 5.00.2184.1)


[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_{03171A57-49B7-4F23-B82F-843F7FA01870}]
SEQPACKET 9
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_{03171A57-49B7-4F23-B82F-843F7FA01870}]
DATAGRAM 9
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_{BA068592-BB15-408D-8679-7AE7A1A70C17}]
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\Nbf_{BA068592-BB15-408D-8679-7AE7A1A70C17}]
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\Nbf_{82B48370-8F39-46C4-82FF-711F34A409D9}]
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\Nbf_{82B48370-8F39-46C4-82FF-711F34A409D9}]
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

Name    MSAFD NetBIOS [\Device\Nbf_NdisWanNbfIn{C0609FFC-1A8F-4270-99B6-
B3B4A88A6CB2}] 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_NdisWanNbfIn{C0609FFC-1A8F-4270-99B6-
B3B4A88A6CB2}] 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{FFA2F77C-7D7A-4F01-85C4-
A5904C04B0B5}] SEQPACKET 7
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{FFA2F77C-7D7A-4F01-85C4-
A5904C04B0B5}] DATAGRAM 7
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{139EC01C-5A3B-4B33-B806-
69DDB63FEB7B}] SEQPACKET 8
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{139EC01C-5A3B-4B33-B806-
69DDB63FEB7B}] DATAGRAM 8
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{0749C430-677C-407E-A0EB-
F68FF7C6233E}] 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_NdisWanNbfOut{0749C430-677C-407E-A0EB-
F68FF7C6233E}] 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{43854D49-A884-4B42-9920-
D7C28ACAF779}] 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{43854D49-A884-4B42-9920-
D7C28ACAF779}] 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\NetBT_Tcpip_{03171A57-49B7-4F23-B82F-
843F7FA01870}] 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\NetBT_Tcpip_{03171A57-49B7-4F23-B82F-
843F7FA01870}] 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\NetBT_Tcpip_{82B48370-8F39-46C4-82FF-
711F34A409D9}] SEQPACKET 3
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_{82B48370-8F39-46C4-82FF-
711F34A409D9}] DATAGRAM 3
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_{BA068592-BB15-408D-8679-
7AE7A1A70C17}] 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_{BA068592-BB15-408D-8679-
7AE7A1A70C17}] 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_{D9FA4DEE-6D1B-4DD1-B157-
1C8899DBA3DD}] SEQPACKET 1
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_{D9FA4DEE-6D1B-4DD1-B157-
1C8899DBA3DD}] DATAGRAM 1
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_{C87CC0F0-CAE8-455A-899F-
3EE55F8A4590}] SEQPACKET 2
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_{C87CC0F0-CAE8-455A-899F-
3EE55F8A4590}] DATAGRAM 2
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 5.00.2195.1207
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     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     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    17.01 GB (18,260,086,784 bytes)
Free Space     13.96 GB (14,988,206,080 bytes)
Volume Name    System
Volume Serial Number   E44180E1
```

```
Partition      Disk #0, Partition #0
Partition Size 17.01 GB (18,260,089,344 bytes)
Starting Offset        32256 bytes
Drive Description      Disk drive
Drive Manufacturer     (Standard disk drives)
Drive Model    UNISYS 018200MAG3182LC SCSI Disk Device
Drive BytesPerSector   512
Drive MediaLoaded      True
Drive MediaType        Fixed hard disk media
Drive Partitions       1
Drive SCSIBus  0
Drive SCSILogicalUnit  0
Drive SCSIPort 1
Drive SCSITargetId     0
Drive SectorsPerTrack  63
Drive Size     18268346880 bytes
Drive TotalCylinders   2221
Drive TotalSectors     35680365
Drive TotalTracks      566355
Drive TracksPerCylinder       255

Drive  E:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  F:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  G:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  H:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  I:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  J:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  K:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  L:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  M:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  N:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  O:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
```

```
Volume Serial Number   Not Available

Drive  P:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  Q:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  R:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  S:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  T:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  U:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  V:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  W:
Description    Local Fixed Disk
Compressed     Not Available
File System    Not Available
Size    Not Available
Free Space     Not Available
Volume Name    Not Available
Volume Serial Number   Not Available

Drive  X:
Description    Local Fixed Disk
Compressed     False
File System    NTFS
Size    474.79 GB (509,802,774,528 bytes)
Free Space     98.00 GB (105,224,470,528 bytes)
Volume Name    Backup1
Volume Serial Number   04AD54DD
Partition      Disk #1, Partition #0
Partition Size 474.79 GB (509,802,822,144 bytes)
Starting Offset        32256 bytes
Drive Description      \\.\PHYSICALDRIVE1
Drive Manufacturer     Not Available
Drive Model    Not Available
Drive BytesPerSector   512
Drive MediaLoaded      True
Drive MediaType        Fixed hard disk media
Drive Partitions       1
Drive SCSIBus  4
Drive SCSILogicalUnit  0
Drive SCSIPort 3
Drive SCSITargetId     0
Drive SectorsPerTrack  63
Drive Size     509802854400 bytes
Drive TotalCylinders   61980
Drive TotalSectors     995708700
Drive TotalTracks      15804900
Drive TracksPerCylinder        255

Drive  Y:
Description    Local Fixed Disk
Compressed     False
File System    NTFS
Size    474.79 GB (509,802,774,528 bytes)
Free Space     98.00 GB (105,221,652,480 bytes)
Volume Name    Backup2
Volume Serial Number   105A473D
Partition      Disk #2, Partition #0
Partition Size 474.79 GB (509,802,822,144 bytes)
Starting Offset        32256 bytes
Drive Description      \\.\PHYSICALDRIVE2
Drive Manufacturer     Not Available
Drive Model    Not Available
Drive BytesPerSector   512
```

```
Drive MediaLoaded     True
Drive MediaType       Fixed hard disk media
Drive Partitions      1
Drive SCSIBus  4
Drive SCSILogicalUnit  0
Drive SCSIPort 3
Drive SCSITargetId    1
Drive SectorsPerTrack  63
Drive Size     509802854400 bytes
Drive TotalCylinders   61980
Drive TotalSectors     995708700
Drive TotalTracks      15804900
Drive TracksPerCylinder       255

Drive   B:
Description    Network Connection
Provider Name  \\CMPX8ADMIN\D$


[SCSI]

Item    Value
Name    Mylex eXtremeRAID 2000 Disk Array Controller
Caption Mylex eXtremeRAID 2000 Disk Array Controller
Driver  dac2w2k
Status  OK
PNP Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&3A654C6B&0&4028
Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&3A654C6B&0&4028
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     54
I/O Port       0x3000-0x3FFF
Driver  c:\winnt\system32\drivers\dac2w2k.sys (185584, 9.00-04)

Name    Symbios Logic 896, 22910 PCI SCSI Adapter
Caption Symbios Logic 896, 22910 PCI SCSI Adapter
Driver  sym_hi
Status  OK
PNP Device ID
       PCI\VEN_1000&DEV_000B&SUBSYS_10001000&REV_05\3&267A616A&0&50
Device ID
       PCI\VEN_1000&DEV_000B&SUBSYS_10001000&REV_05\3&267A616A&0&50
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     58
I/O Port       0x1000-0x10FF
Driver  c:\winnt\system32\drivers\sym_hi.sys (21136, 5.00.2134.1)

Name    Symbios Logic 896, 22910 PCI SCSI Adapter
Caption Symbios Logic 896, 22910 PCI SCSI Adapter
Driver  sym_hi
Status  OK
PNP Device ID
       PCI\VEN_1000&DEV_000B&SUBSYS_10001000&REV_05\3&267A616A&0&51
Device ID
       PCI\VEN_1000&DEV_000B&SUBSYS_10001000&REV_05\3&267A616A&0&51
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     18
I/O Port       0x1400-0x14FF
Driver  c:\winnt\system32\drivers\sym_hi.sys (21136, 5.00.2134.1)

Name    Mylex eXtremeRAID 2000 Disk Array Controller
Caption Mylex eXtremeRAID 2000 Disk Array Controller
Driver  dac2w2k
Status  OK
PNP Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&2A769D37&0&4030
Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&2A769D37&0&4030
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     40
I/O Port       0x4000-0x5FFF
Driver  c:\winnt\system32\drivers\dac2w2k.sys (185584, 9.00-04)

Name    Mylex eXtremeRAID 2000 Disk Array Controller
Caption Mylex eXtremeRAID 2000 Disk Array Controller
Driver  dac2w2k
Status  OK
PNP Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&2FBC1DEA&0&4038
Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&2FBC1DEA&0&4038
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     36
I/O Port       0x5000-0x5FFF
Driver  c:\winnt\system32\drivers\dac2w2k.sys (185584, 9.00-04)

Name    Mylex eXtremeRAID 2000 Disk Array Controller
Caption Mylex eXtremeRAID 2000 Disk Array Controller
Driver  dac2w2k
Status  OK
PNP Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&8C49857&0&4020
Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&8C49857&0&4020
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     32
I/O Port       0x6000-0x7FFF
Driver  c:\winnt\system32\drivers\dac2w2k.sys (185584, 9.00-04)

Name    Mylex eXtremeRAID 2000 Disk Array Controller
Caption Mylex eXtremeRAID 2000 Disk Array Controller
Driver  dac2w2k
Status  OK
```

```
PNP Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&375C4928&0&4028
Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&375C4928&0&4028
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     28
I/O Port       0x7000-0x7FFF
Driver  c:\winnt\system32\drivers\dac2w2k.sys (185584, 9.00-04)

Name    Mylex eXtremeRAID 2000 Disk Array Controller
Caption Mylex eXtremeRAID 2000 Disk Array Controller
Driver  dac2w2k
Status  OK
PNP Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&1B89A02&0&4020
Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&1B89A02&0&4020
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     24
I/O Port       0x8000-0x9FFF
Driver  c:\winnt\system32\drivers\dac2w2k.sys (185584, 9.00-04)

Name    Mylex eXtremeRAID 2000 Disk Array Controller
Caption Mylex eXtremeRAID 2000 Disk Array Controller
Driver  dac2w2k
Status  OK
PNP Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&1CDF5718&0&4028
Device ID
       PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&1CDF5718&0&4028
Device Map     Not Available
Index   Not Available
Max Number Controlled  Not Available
IRQ Number     20
I/O Port       0x9000-0x9FFF
Driver  c:\winnt\system32\drivers\dac2w2k.sys (185584, 9.00-04)


[Printing]

Name    Port Name      Server Name
No printing information

[Problem Devices]

Device  PNP Device ID  Error Code
Intel(R) PRO/100+ Dual Port Server Adapter #2
       PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&27BD01E1&0&2820
       22

[USB]

Device  PNP Device ID
Intel 82371AB/EB PCI to USB Universal Host Controller
       PCI\VEN_8086&DEV_7112&SUBSYS_00000000&REV_01\3&267A616A&0&7A
USB Root Hub   USB\ROOT_HUB\4&B5B4E1B&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
       Disabled       Stopped OK      Ignore  False   False
acpi    Microsoft ACPI Driver  c:\winnt\system32\drivers\acpi.sys    Kernel
Driver  True    Boot    Running OK      Normal  False   True
acpiec  ACPIEC  c:\winnt\system32\drivers\acpiec.sys  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
adpu160m       adpu160m       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
afd     AFD Networking Support Environment
       c:\winnt\system32\drivers\afd.sys     Kernel Driver  True    Auto
       Running OK      Normal  False   True
aic116x aic116x Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
aic78u2 aic78u2 Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
aic78xx aic78xx Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
ami0nt  ami0nt  Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
asc     asc     Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
asc3550 asc3550 Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
asyncmac       RAS Asynchronous Media Driver
       c:\winnt\system32\drivers\asyncmac.sys       Kernel Driver  False
       Manual  Stopped OK      Normal  False   False
atapi   Standard IDE/ESDI Hard Disk Controller
       c:\winnt\system32\drivers\atapi.sys   Kernel Driver  True    Boot
       Running OK      Normal  False   True
atdisk  Atdisk  Not Available  Kernel Driver  False   Disabled
       Stopped OK      Ignore  False   False
atmarpc ATM ARP Client Protocol
       c:\winnt\system32\drivers\atmarpc.sys Kernel Driver  False   Manual
       Stopped OK      Normal  False   False
audstub Audio Stub Driver      c:\winnt\system32\drivers\audstub.sys Kernel
Driver  True    Manual  Running OK      Normal  False   True
beep    Beep    c:\winnt\system32\drivers\beep.sys    Kernel Driver  True
       System  Running OK      Normal  False   True
cdaudio Cdaudio c:\winnt\system32\drivers\cdaudio.sys Kernel Driver  False
       System  Stopped OK      Ignore  False   False
cdfs    Cdfs    c:\winnt\system32\drivers\cdfs.sys    File System Driver
       True    Disabled       Running OK      Normal  False   True
cdrom   CD-ROM Driver  c:\winnt\system32\drivers\cdrom.sys   Kernel Driver
       True    System  Running OK      Normal  False   True
changer Changer Not Available  Kernel Driver  False   System  Stopped OK
       Ignore  False   False
cirrus  cirrus  c:\winnt\system32\drivers\cirrus.sys  Kernel Driver  True
       Manual  Running OK      Ignore  False   True
```

```
cpqarray       Cpqarray       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
cpqarry2       cpqarry2       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
cpqfcalm       cpqfcalm       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
dac2w2k dac2w2k c:\winnt\system32\drivers\dac2w2k.sys Kernel Driver  True
       Boot    Running OK      Normal  False   True
dac960nt       dac960nt       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
deckzpsx       deckzpsx       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
dfsdriver      DfsDriver      c:\winnt\system32\drivers\dfs.sys    File
System Driver  True    Boot    Running OK      Normal  False   True
disk    Disk Driver    c:\winnt\system32\drivers\disk.sys    Kernel Driver
       True    Boot    Running OK      Normal  False   True
diskperf       Diskperf       c:\winnt\system32\drivers\diskperf.sys
       Kernel Driver  False   Disabled        Stopped OK      Normal  False
       False
dmboot  dmboot  c:\winnt\system32\drivers\dmboot.sys  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
dmio    Logical Disk Manager Driver    c:\winnt\system32\drivers\dmio.sys
       Kernel Driver  True    Boot    Running OK      Normal  False   True
dmload  dmload  c:\winnt\system32\drivers\dmload.sys  Kernel Driver  True
       Boot    Running OK      Normal  False   True
e100b   Intel PRO Adapter Driver
       c:\winnt\system32\drivers\e100bnt5.sys       Kernel Driver  True
       Manual  Running OK      Normal  False   True
efs     EFS     c:\winnt\system32\drivers\efs.sys     File System Driver
       True    Disabled       Running OK      Normal  False   True
fastfat Fastfat c:\winnt\system32\drivers\fastfat.sys File System Driver
       True    Disabled       Running OK      Normal  False   True
fdc     Floppy Disk Controller Driver c:\winnt\system32\drivers\fdc.sys
       Kernel Driver  True    Manual  Running OK      Normal  False   True
fireport       fireport       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
flpydisk       Floppy Disk Driver
       c:\winnt\system32\drivers\flpydisk.sys       Kernel Driver  True
       Manual  Running OK      Normal  False   True
ftdisk  Volume Manager Driver  c:\winnt\system32\drivers\ftdisk.sys  Kernel
Driver  True    Boot    Running OK      Normal  False   True
gamdrv  gamdrv  c:\winnt\system32\drivers\gamdrv.sys  Kernel Driver  True
       Boot    Running OK      Normal  False   True
gnindis cLAN NDIS Driver       c:\winnt\system32\drivers\gnindis.sys Kernel
Driver  True    Auto    Running OK      Normal  False   True
gninvipl       cLAN VIPL Driver
       c:\winnt\system32\drivers\gninvipl.sys       Kernel Driver  False
       Manual  Stopped OK      Normal  False   False
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
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
intelide       IntelIde       c:\winnt\system32\drivers\intelide.sys
       Kernel Driver  True    Boot    Running OK      Normal  False   True
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
macdisk macdisk c:\winnt\system32\drivers\mac2w2k.sys Kernel Driver  True
       Boot    Running OK      Normal  False   True
mnmdd   mnmdd   c:\winnt\system32\drivers\mnmdd.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
mraid35x       mraid35x       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
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\netdtect.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    Disabled        Running OK      Normal  False   True
null    Null    c:\winnt\system32\drivers\null.sys    Kernel Driver  True
        System  Running OK      Normal  False   True
nwlnkflt        IPX Traffic Filter Driver
        c:\winnt\system32\drivers\nwlnkflt.sys       Kernel Driver  False
        Manual  Stopped OK      Normal  False   False
nwlnkfwd        IPX Traffic Forwarder Driver
        c:\winnt\system32\drivers\nwlnkfwd.sys       Kernel Driver  False
        Manual  Stopped OK      Normal  False   False
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  Not Available   Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
pcmcia  Pcmcia  c:\winnt\system32\drivers\pcmcia.sys  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
pdcomp  PDCOMP  Not Available   Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False
pdframe PDFRAME Not Available   Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False
pdreli  PDRELI  Not Available   Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False
pdrframe        PDRFRAME        Not Available   Kernel Driver  False   Manual
        Stopped OK      Ignore  False   False
pptpminiport    WAN Miniport (PPTP)
        c:\winnt\system32\drivers\raspptp.sys Kernel Driver  True    Manual
        Running OK      Normal  False   True
ptilink Direct Parallel Link Driver
        c:\winnt\system32\drivers\ptilink.sys Kernel Driver  True    Manual
        Running OK      Normal  False   True
ql1080  ql1080  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
ql10wnt Ql10wnt Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
ql1240  ql1240  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
ql2100  ql2100  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
rasacd  Remote Access Auto Connection Driver
        c:\winnt\system32\drivers\rasacd.sys  Kernel Driver  True    System
        Running OK      Normal  False   True
rasl2tp WAN Miniport (L2TP)    c:\winnt\system32\drivers\rasl2tp.sys Kernel
Driver  True    Manual  Running OK      Normal  False   True
raspti  Direct Parallel        c:\winnt\system32\drivers\raspti.sys  Kernel
Driver  True    Manual  Running OK      Normal  False   True
rca     Microsoft Streaming Network Raw Channel Access
        c:\winnt\system32\drivers\rca.sys     Kernel Driver  False   Manual
        Stopped OK      Normal  False   False
rdbss   Rdbss   c:\winnt\system32\drivers\rdbss.sys   File System Driver
        True    System  Running OK      Normal  False   True
rdpwd   RDPWD   c:\winnt\system32\drivers\rdpwd.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
redbook Digital CD Audio Playback Filter Driver
        c:\winnt\system32\drivers\redbook.sys Kernel Driver  False   System
        Stopped OK      Normal  False   False
serenum Serenum Filter Driver  c:\winnt\system32\drivers\serenum.sys Kernel
Driver  True    Manual  Running OK      Normal  False   True
serial  Serial port driver     c:\winnt\system32\drivers\serial.sys  Kernel
Driver  True    System  Running OK      Ignore  False   True
sfloppy Sfloppy c:\winnt\system32\drivers\sfloppy.sys Kernel Driver  False
        System  Stopped OK      Ignore  False   False
sglfb   sglfb   Not Available  Kernel Driver  False   System  Stopped OK
        Normal  False   False
simbad  Simbad  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
srv     Srv     c:\winnt\system32\drivers\srv.sys     File System Driver
        True    Manual  Running OK      Normal  False   True
swenum  Software Bus Driver    c:\winnt\system32\drivers\swenum.sys  Kernel
Driver  True    Manual  Running OK      Normal  False   True
symc810 symc810 Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
symc8xx symc8xx Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
sym_hi  sym_hi  c:\winnt\system32\drivers\sym_hi.sys  Kernel Driver  True
        Boot    Running OK      Normal  False   True
sysdrv  Sysdrv  c:\winnt\system32\drivers\sysdrv.sys  Kernel Driver  True
        Auto    Running OK      Normal  False   True
tcpip   TCP/IP Protocol Driver c:\winnt\system32\drivers\tcpip.sys   Kernel
Driver  True    System  Running OK      Normal  False   True
tdasync TDASYNC c:\winnt\system32\drivers\tdasync.sys Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdipx   TDIPX   c:\winnt\system32\drivers\tdipx.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdnetb  TDNETB  c:\winnt\system32\drivers\tdnetb.sys  Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdpipe  TDPIPE  c:\winnt\system32\drivers\tdpipe.sys  Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdspx   TDSPX   c:\winnt\system32\drivers\tdspx.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdtcp   TDTCP   c:\winnt\system32\drivers\tdtcp.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
termdd  Terminal Device Driver c:\winnt\system32\drivers\termdd.sys  Kernel
Driver  False   Disabled        Stopped OK      Normal  False   False
```

```
tga     tga     Not Available  Kernel Driver  False   System  Stopped OK
        Ignore  False   False
udfs    Udfs    c:\winnt\system32\drivers\udfs.sys    File System Driver
        False   Disabled        Stopped OK      Normal  False   False
uhcd    Microsoft USB Universal Host Controller Driver
        c:\winnt\system32\drivers\uhcd.sys    Kernel Driver  True    Manual
        Running OK      Normal  False   True
update  Microcode Update Driver       c:\winnt\system32\drivers\update.sys
        Kernel Driver  True    Manual  Running OK      Normal  False   True
usbhub  Microsoft USB Standard Hub Driver
        c:\winnt\system32\drivers\usbhub.sys  Kernel Driver  True    Manual
        Running OK      Normal  False   True
vgasave VgaSave c:\winnt\system32\drivers\vga.sys     Kernel Driver  False
        System  Stopped OK      Ignore  False   False
wanarp  Remote Access IP ARP Driver   c:\winnt\system32\drivers\wanarp.sys
        Kernel Driver  True    Manual  Running OK      Normal  False   True
wdica   WDICA   Not Available  Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False

[Environment Variables]

Variable        Value   User Name
ComSpec %SystemRoot%\system32\cmd.exe <SYSTEM>
Os2LibPath      %SystemRoot%\system32\os2\dll;        <SYSTEM>
Path
        %SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;C:\P
rogram Files\Microsoft SQL Server\80\Tools\BINN      <SYSTEM>
windir  %SystemRoot%    <SYSTEM>
OS      Windows_NT      <SYSTEM>
PROCESSOR_ARCHITECTURE x86      <SYSTEM>
PROCESSOR_LEVEL         6       <SYSTEM>
PROCESSOR_IDENTIFIER    x86 Family 6 Model 10 Stepping 1, GenuineIntel
        <SYSTEM>
PROCESSOR_REVISION      0a01    <SYSTEM>
NUMBER_OF_PROCESSORS    8       <SYSTEM>
PATHEXT .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH     <SYSTEM>
TEMP    %SystemRoot%\TEMP       <SYSTEM>
TMP     %SystemRoot%\TEMP       <SYSTEM>
TEMP    %USERPROFILE%\Local Settings\Temp     CAPRICORN8\Administrator
TMP     %USERPROFILE%\Local Settings\Temp     CAPRICORN8\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 Unknown
        Unknown

[Network Connections]

Local Name      Remote Name     Type    Status  User Name
B:      \\CMPX8ADMIN\D$         Disk    OK      CAPRICORN8\Administrator

[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       Not Available   Unknown Unknown Unknown
system  Not Available  8       8       0       1413120 Not Available
        Unknown Unknown Unknown
smss.exe        c:\winnt\system32\smss.exe      188     11      204800
        1413120 10/30/2000 7:21:44 AM  5.00.2195.31    44.27 KB (45,328
bytes)  8/8/2000 5:00:00 AM
csrss.exe       Not Available  212     13      Not Available   Not Available
        10/30/2000 7:21:49 AM  Unknown Unknown Unknown
winlogon.exe    c:\winnt\system32\winlogon.exe         232     13      204800
        1413120 10/30/2000 7:21:51 AM  5.00.2195.1600 172.77 KB (176,912
bytes)  8/8/2000 5:00:00 AM
services.exe    c:\winnt\system32\services.exe         260     9       204800
        1413120 10/30/2000 7:21:52 AM  5.00.2134.1    86.77 KB (88,848
bytes)  8/8/2000 5:00:00 AM
lsass.exe       c:\winnt\system32\lsass.exe    272     13      204800
        1413120 10/30/2000 7:21:52 AM  5.00.2195.1620 32.77 KB (33,552
bytes)  8/8/2000 5:00:00 AM
gnconmgr.exe    c:\winnt\system32\gnconmgr.exe         400     8       204800
        1413120 10/30/2000 7:21:55 AM  4.01.00 128.05 KB (131,119 bytes)
        8/16/1999 7:09:37 AM
procconsvc.exe c:\winnt\system32\procconsvc.exe      452     8       204800
        1413120 10/30/2000 7:21:55 AM  1, 0, 1, 23    72.27 KB (74,000
bytes)  10/3/2000 8:36:28 AM
regsvc.exe      c:\winnt\system32\regsvc.exe  476     8       204800
        1413120 10/30/2000 7:21:56 AM  5.00.2195.31    65.27 KB (66,832
bytes)  8/8/2000 5:00:00 AM
procconmd8.exe c:\winnt\system32\procconmd8.exe      472     8       204800
        1413120 10/30/2000 7:21:56 AM  1, 0, 1, 23    24.27 KB (24,848
bytes)  10/3/2000 8:36:28 AM
svchost.exe     c:\winnt\system32\svchost.exe 500     8       204800
        1413120 10/30/2000 7:21:56 AM  5.00.2134.1    7.77 KB (7,952 bytes)
        8/8/2000 5:00:00 AM
snmp.exe        c:\winnt\system32\snmp.exe     544     8       204800
        1413120 10/30/2000 7:21:56 AM  5.00.2173.1    29.77 KB (30,480
bytes)  10/4/2000 3:06:01 PM
svchost.exe     c:\winnt\system32\svchost.exe 676     8       204800
        1413120 10/30/2000 7:22:01 AM  5.00.2134.1    7.77 KB (7,952 bytes)
        8/8/2000 5:00:00 AM
winmgmt.exe     c:\winnt\system32\wbem\winmgmt.exe     696     8       204800
        1413120 10/30/2000 7:22:01 AM  1.50.1085.0009 192.08 KB (196,685
bytes)  8/8/2000 5:00:00 AM
explorer.exe    c:\winnt\explorer.exe  784     8       204800  1413120
        10/30/2000 7:22:11 AM  5.00.3103.1000 237.27 KB (242,960 bytes)
        8/8/2000 5:00:00 AM
svchost.exe     c:\winnt\system32\svchost.exe 960     8       204800
        1413120 10/30/2000 7:22:17 AM  5.00.2134.1    7.77 KB (7,952 bytes)
        8/8/2000 5:00:00 AM
cmd.exe c:\winnt\system32\cmd.exe      612     8       204800  1413120
        10/30/2000 7:22:27 AM  5.00.2195.1600 230.77 KB (236,304 bytes)
        8/8/2000 5:00:00 AM
sqlservr.exe    c:\program files\microsoft sql
server\mssql\binn\sqlservr.exe         812     13      204800  1413120
```

```
       10/30/2000 7:22:29 AM  2000.080.0194.00      7.10 MB (7,442,493
bytes)  10/4/2000 2:36:12 PM
mmc.exe c:\winnt\system32\mmc.exe     1032    8       204800  1413120
       10/30/2000 7:25:04 AM  5.00.2153.1    589.27 KB (603,408 bytes)
       8/8/2000 5:00:00 AM
isqlw.exe      c:\program files\microsoft sql
server\80\tools\binn\isqlw.exe        324     8       204800  1413120
       10/30/2000 10:41:04 AM 2000.080.0194.00      344.06 KB (352,319
bytes)  10/4/2000 2:36:48 PM
mmc.exe c:\winnt\system32\mmc.exe     892     8       204800  1413120
       10/30/2000 2:46:13 PM  5.00.2153.1    589.27 KB (603,408 bytes)
       8/8/2000 5:00:00 AM
rsvp.exe       c:\winnt\system32\rsvp.exe    1164    8       204800
       1413120 10/30/2000 2:48:17 PM  5.00.2167.1    172.77 KB (176,912
bytes)  8/8/2000 5:00:00 AM

[Loaded Modules]

Name    Version Size    File Date      Manufacturer   Path
traffic.dll    5.00.2139.1    30.77 KB (31,504 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\traffic.dll
rsvp.exe       5.00.2167.1    172.77 KB (176,912 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rsvp.exe
wbemprox.dll   1.50.1085.0015 40.08 KB (41,040 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\wbemprox.dll
rassapi.dll    5.00.2188.1    14.27 KB (14,608 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rassapi.dll
adsnt.dll      5.00.2195.1600 194.27 KB (198,928 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\adsnt.dll
dbghelp.dll    5.00.2195.1620 159.27 KB (163,088 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\dbghelp.dll
localsec.dll   5.00.2195.1340 227.27 KB (232,720 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\localsec.dll
devmgr.dll     5.00.2166.1    215.77 KB (220,944 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\devmgr.dll
filemgmt.dll   5.00.2134.1    287.27 KB (294,160 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\filemgmt.dll
smlogcfg.dll   5.00.2163.1    273.27 KB (279,824 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\smlogcfg.dll
cabinet.dll    5.00.2147.1    54.77 KB (56,080 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\cabinet.dll
msinfo32.dll   5.00.2177.1    312.27 KB (319,760 bytes)     10/3/2000
3:38:21 PM     Microsoft Corporation  c:\program files\common
files\microsoft shared\msinfo\msinfo32.dll
riched20.dll   5.30.23.1203   421.27 KB (431,376 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\riched20.dll
riched32.dll   5.00.2134.1    3.77 KB (3,856 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\riched32.dll
els.dll 5.00.2175.1    151.27 KB (154,896 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\els.dll
ntmsmgr.dll    1,0,0,1 427.77 KB (438,032 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation and HighGround Systems, Inc.
       c:\winnt\system32\ntmsmgr.dll
mmfutil.dll    1.50.1085.0000 32.06 KB (32,829 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mmfutil.dll
logdrive.dll   1.50.1085.0000 200.06 KB (204,863 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\logdrive.dll
dfrgres.dll    5.00.2150.1    27.50 KB (28,160 bytes)       8/8/2000
5:00:00 AM     Executive Software International, Inc.
       c:\winnt\system32\dfrgres.dll
dfrgsnap.dll   5.00.2195.31   41.77 KB (42,768 bytes)       8/8/2000
5:00:00 AM     Executive Software International, Inc.
       c:\winnt\system32\dfrgsnap.dll
dmdskres.dll   2195.1600.297.3       119.50 KB (122,368 bytes)
       8/8/2000 5:00:00 AM    Microsoft Corp., VERITAS Software
       c:\winnt\system32\dmdskres.dll
dmutil.dll     2195.23.297.2  42.27 KB (43,280 bytes)       8/8/2000
5:00:00 AM     VERITAS Software Corp. c:\winnt\system32\dmutil.dll
ntmsapi.dll    5.00.1948.1    50.27 KB (51,472 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntmsapi.dll
dmdskmgr.dll   2195.1600.297.3       160.27 KB (164,112 bytes)
       8/8/2000 5:00:00 AM    Microsoft Corp., VERITAS Software
       c:\winnt\system32\dmdskmgr.dll
mycomput.dll   5.00.2134.1    107.77 KB (110,352 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mycomput.dll
dbmslpcn.dll   2000.080.0194.00       28.06 KB (28,734 bytes)
       10/4/2000 2:36:17 PM   Microsoft Corporation
       c:\winnt\system32\dbmslpcn.dll
dbnetlib.dll   2000.080.0194.00       84.06 KB (86,082 bytes)
       10/4/2000 2:35:43 PM   Microsoft Corporation
       c:\winnt\system32\dbnetlib.dll
sqllex.dll     2000.080.0194.00       148.06 KB (151,616 bytes)
       10/4/2000 2:37:13 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\sqllex.dll
odbccp32.dll   3.520.6526.0   100.27 KB (102,672 bytes)     10/4/2000
2:35:34 PM     Microsoft Corporation  c:\winnt\system32\odbccp32.dll
sqlsrv32.rll    2000.080.0194.00      88.00 KB (90,112 bytes)
       10/4/2000 2:35:44 PM   Microsoft Corporation
       c:\winnt\system32\sqlsrv32.rll
sqlsrv32.dll    2000.080.0194.00      460.08 KB (471,119 bytes)
       10/4/2000 2:35:44 PM   Microsoft Corporation
       c:\winnt\system32\sqlsrv32.dll
isqlw.rll      2000.080.0194.00       240.00 KB (245,760 bytes)
       10/4/2000 2:36:48 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\resources\1033\isqlw.rll
sqlqry.rll     2000.080.0194.00       180.00 KB (184,320 bytes)
       10/4/2000 2:36:49 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\resources\1033\sqlqry.rll
pfutil80.rll   2000.080.0194.00       144.00 KB (147,456 bytes)
       10/4/2000 2:37:15 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\resources\1033\pfutil80.rll
pfclnt80.rll   2000.080.0194.00       28.00 KB (28,672 bytes)
       10/4/2000 2:37:11 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\resources\1033\pfclnt80.rll
semsfc.rll     2000.080.0194.00       24.00 KB (24,576 bytes)
       10/4/2000 2:37:14 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\resources\1033\semsfc.rll
sqlgui.rll     2000.080.0194.00       56.00 KB (57,344 bytes)
       10/4/2000 2:37:14 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\resources\1033\sqlgui.rll
sqlsvc.rll     2000.080.0194.00       24.00 KB (24,576 bytes)
       10/4/2000 2:37:11 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\resources\1033\sqlsvc.rll
odbcint.dll    3.520.6526.0   88.00 KB (90,112 bytes)       10/4/2000
2:35:34 PM     Microsoft Corporation  c:\winnt\system32\odbcint.dll
```

```
pfclnt80.dll   2000.080.0194.00      404.06 KB (413,762 bytes)
       10/4/2000 2:37:11 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\pfclnt80.dll
semsfc.dll     2000.080.0194.00      224.06 KB (229,440 bytes)
       10/4/2000 2:37:12 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\semsfc.dll
pfutil80.dll   2000.080.0194.00      268.06 KB (274,498 bytes)
       10/4/2000 2:37:14 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\pfutil80.dll
sqlqry.dll     2000.080.0194.00      392.06 KB (401,472 bytes)
       10/4/2000 2:36:48 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\sqlqry.dll
imm32.dll      5.00.2195.1387 94.77 KB (97,040 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\imm32.dll
odbcbcp.dll     2000.080.0194.00     28.07 KB (28,742 bytes)
       10/4/2000 2:35:44 PM   Microsoft Corporation
       c:\winnt\system32\odbcbcp.dll
sqlsvc.dll     2000.080.0194.00      92.06 KB (94,272 bytes)
       10/4/2000 2:37:11 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\sqlsvc.dll
odbc32.dll     3.520.6526.0   216.27 KB (221,456 bytes)     10/4/2000
2:35:34 PM     Microsoft Corporation  c:\winnt\system32\odbc32.dll
w95scm.dll     2000.080.0194.00      48.06 KB (49,216 bytes)
       10/4/2000 2:37:10 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\w95scm.dll
sqlgui.dll     2000.080.0194.00      444.06 KB (454,720 bytes)
       10/4/2000 2:37:11 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\sqlgui.dll
sqlresld.dll   2000.080.0194.00      28.06 KB (28,738 bytes)
       10/4/2000 2:37:11 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\sqlresld.dll
sqlunirl.dll   2000.080.0194.00      176.06 KB (180,290 bytes)
       8/6/2000 1:51:56 AM    Microsoft Corporation
       c:\winnt\system32\sqlunirl.dll
isqlw.exe      2000.080.0194.00      344.06 KB (352,319 bytes)
       10/4/2000 2:36:48 PM   Microsoft Corporation  c:\program
files\microsoft sql server\80\tools\binn\isqlw.exe
mlang.dll      5.00.3103.1000 510.77 KB (523,024 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mlang.dll
tapiperf.dll   5.00.2195.1600 6.27 KB (6,416 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\tapiperf.dll
rsvpperf.dll   5.00.2167.1    10.77 KB (11,024 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rsvpperf.dll
rasctrs.dll    5.00.2188.1    12.27 KB (12,560 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasctrs.dll
perfctrs.dll   5.00.2134.1    40.77 KB (41,744 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\perfctrs.dll
sqlctr80.dll   2000.080.0194.00      32.06 KB (32,827 bytes)
       10/4/2000 2:36:35 PM   Microsoft Corporation
       c:\progra~1\micros~2\mssql\binn\sqlctr80.dll
gni_perf.dll   Not Available  48.00 KB (49,152 bytes)       2/18/1999
12:26:45 PM    Not Available  c:\winnt\system32\gni_perf.dll
faxperf.dll    5.00.2148.1    6.27 KB (6,416 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\faxperf.dll
olepro32.dll   5.0.4514       160.27 KB (164,112 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\olepro32.dll
pdh.dll 5.00.2195.1600 143.27 KB (146,704 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\pdh.dll
sysmon.ocx     5.00.2163.1    167.77 KB (171,792 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\sysmon.ocx
mmcndmgr.dll   5.00.2178.1    815.27 KB (834,832 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mmcndmgr.dll
mmc.exe 5.00.2153.1    589.27 KB (603,408 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\mmc.exe
ssmsgnet.dll   2000.080.0194.00      32.06 KB (32,830 bytes)
       10/4/2000 2:37:19 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\ssmsgnet.dll
ssmslpcn.dll   2000.080.0194.00      28.06 KB (28,734 bytes)
       10/4/2000 2:36:16 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\ssmslpcn.dll
security.dll   5.00.2195.1600 5.77 KB (5,904 bytes) 8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\security.dll
ssnmpn70.dll   2000.080.0194.00      24.06 KB (24,638 bytes)
       10/4/2000 2:36:16 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\ssnmpn70.dll
ssnetlib.dll   2000.080.0194.00      84.06 KB (86,078 bytes)
       10/4/2000 2:36:15 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\ssnetlib.dll
resutils.dll   5.00.2195.1613 39.77 KB (40,720 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\resutils.dll
clusapi.dll    5.00.2195.1613 54.27 KB (55,568 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\clusapi.dll
mtxclu.dll     1999.9.3421.3  50.27 KB (51,472 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mtxclu.dll
msdtcprx.dll   2000.2.3449.0  625.77 KB (640,784 bytes)     10/3/2000
8:36:11 AM     Microsoft Corporation  c:\winnt\system32\msdtcprx.dll
xolehlp.dll    1999.9.3421.3  17.27 KB (17,680 bytes)       10/3/2000
8:36:11 AM     Microsoft Corporation  c:\winnt\system32\xolehlp.dll
sqlevn70.rll   2000.080.0194.00      28.00 KB (28,672 bytes)
       10/4/2000 2:36:16 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\resources\1033\sqlevn70.rll
msvcirt.dll    6.10.8637.0    76.05 KB (77,878 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msvcirt.dll
sqlsort.dll    2000.080.0194.00      576.06 KB (589,885 bytes)
       10/4/2000 2:36:15 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\sqlsort.dll
ums.dll 2000.080.0194.00       48.06 KB (49,210 bytes)       10/4/2000
2:36:14 PM     Microsoft Corporation  c:\program files\microsoft sql
server\mssql\binn\ums.dll
opends60.dll   2000.080.0194.00      24.06 KB (24,639 bytes)
       10/4/2000 2:36:14 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\opends60.dll
sqlservr.exe   2000.080.0194.00      7.10 MB (7,442,493 bytes)
       10/4/2000 2:36:12 PM   Microsoft Corporation  c:\program
files\microsoft sql server\mssql\binn\sqlservr.exe
cmd.exe 5.00.2195.1600 230.77 KB (236,304 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\cmd.exe
h323.tsp       5.00.2195.1600 248.27 KB (254,224 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\h323.tsp
ipconf.tsp     5.00.2195.1600 10.77 KB (11,024 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ipconf.tsp
ndptsp.tsp     5.00.2195.1600 38.27 KB (39,184 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ndptsp.tsp
kmddsp.tsp     5.00.2195.1600 18.27 KB (18,704 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\kmddsp.tsp
uniplat.dll    5.00.2151.1    13.77 KB (14,096 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\uniplat.dll
```

```
unimdm.tsp     5.00.2175.1    196.77 KB (201,488 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\unimdm.tsp
tapisrv.dll    5.00.2195.1600 168.77 KB (172,816 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\tapisrv.dll
wzcab2.dll     2, 0, 0, 0     20.50 KB (20,992 bytes)       9/22/1998
7:00:00 AM     Nico Mak Computing, Inc.      c:\progra~1\winzip\wzcab2.dll
crtdll.dll     4.00   145.77 KB (149,264 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\crtdll.dll
wzshlext.dll   Not Available  33.00 KB (33,792 bytes)       9/22/1998
7:00:00 AM     Not Available  c:\progra~1\winzip\wzshlext.dll
wininet.dll    5.00.3103.1000 456.77 KB (467,728 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wininet.dll
shdoclc.dll    5.00.3103.1000 324.50 KB (332,288 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\shdoclc.dll
urlmon.dll     5.00.3103.1000 440.77 KB (451,344 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\urlmon.dll
ntshrui.dll    5.00.2134.1    46.77 KB (47,888 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntshrui.dll
linkinfo.dll   5.00.2195.1387 16.77 KB (17,168 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\linkinfo.dll
browselc.dll   5.00.3103.1000 34.50 KB (35,328 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\browselc.dll
mydocs.dll     5.00.3103.1000 57.77 KB (59,152 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mydocs.dll
faxshell.dll   5.00.2134.1    8.27 KB (8,464 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\faxshell.dll
msacm32.dll    5.00.2134.1    65.27 KB (66,832 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msacm32.dll
avifil32.dll   5.00.2134.1    76.27 KB (78,096 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\avifil32.dll
msvfw32.dll    5.00.2134.1    113.77 KB (116,496 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msvfw32.dll
docprop2.dll   5.00.2195.1387 308.77 KB (316,176 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\docprop2.dll
hhsetup.dll    4.74.8702      66.27 KB (67,856 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\hhsetup.dll
mfc42u.dll     6.00.8665.0    972.05 KB (995,384 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mfc42u.dll
mmcshext.dll   5.00.2153.1    24.27 KB (24,848 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mmcshext.dll
msi.dll 1.11.1314.0    1.72 MB (1,798,928 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\msi.dll
powrprof.dll   5.00.3103.1000 13.27 KB (13,584 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\powrprof.dll
batmeter.dll   5.00.2920.0000 20.27 KB (20,752 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\batmeter.dll
stobject.dll   5.00.2195.1387 79.27 KB (81,168 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\stobject.dll
webcheck.dll   5.00.3103.1000 251.77 KB (257,808 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\webcheck.dll
browseui.dll   5.00.3103.1000 788.77 KB (807,696 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\browseui.dll
shdocvw.dll    5.00.3103.1000 1.05 MB (1,104,144 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\shdocvw.dll
explorer.exe   5.00.3103.1000 237.27 KB (242,960 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\explorer.exe
rapilib.dll    5.00.2167.1    25.27 KB (25,872 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rapilib.dll
rsvpsp.dll     5.00.2167.1    74.77 KB (76,560 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rsvpsp.dll
provthrd.dll   1.50.1085.0000 68.07 KB (69,708 bytes)       10/3/2000
3:38:11 PM     Microsoft Corporation  c:\winnt\system32\wbem\provthrd.dll
ntevt.dll      1.50.1085.0000 192.06 KB (196,669 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\ntevt.dll
framedyn.dll   1.50.1085.0000 164.05 KB (167,992 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\framedyn.dll
cimwin32.dll   1.50.1085.0016 1.02 MB (1,073,232 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\cimwin32.dll
wbemsvc.dll    1.50.1085.0007 40.07 KB (41,036 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\wbemsvc.dll
wbemess.dll    1.50.1085.0007 364.07 KB (372,804 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\wbemess.dll
fastprox.dll   1.50.1085.0007 144.08 KB (147,536 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\fastprox.dll
wbemcore.dll   1.50.1085.0008 628.07 KB (643,140 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\wbemcore.dll
wbemcomn.dll   1.50.1085.0007 692.07 KB (708,675 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\wbemcomn.dll
winmgmt.exe    1.50.1085.0009 192.08 KB (196,685 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\winmgmt.exe
ipbootp.dll    5.00.2195.1284 33.77 KB (34,576 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ipbootp.dll
rastls.dll     5.00.2188.1    47.77 KB (48,912 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rastls.dll
raschap.dll    5.00.2175.1    34.77 KB (35,600 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\raschap.dll
ntlsapi.dll    5.00.2134.1    6.77 KB (6,928 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\ntlsapi.dll
rasppp.dll     5.00.2195.1203 192.27 KB (196,880 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasppp.dll
rastapi.dll    5.00.2188.1    52.27 KB (53,520 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rastapi.dll
rasdlg.dll     5.00.2194.1    514.27 KB (526,608 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasdlg.dll
netcfgx.dll    5.00.2195.1618 533.77 KB (546,576 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netcfgx.dll
rasmans.dll    5.00.2195.27   146.77 KB (150,288 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasmans.dll
wmi.dll 5.00.2195.1600 6.27 KB (6,416 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\wmi.dll
netshell.dll   5.00.2195.1600 456.77 KB (467,728 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netshell.dll
netman.dll     5.00.2195.1600 89.27 KB (91,408 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netman.dll
sens.dll       5.00.2163.1    36.77 KB (37,648 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\sens.dll
txfaux.dll     1999.9.3422.24 341.27 KB (349,456 bytes)     10/3/2000
8:36:11 AM     Microsoft Corporation  c:\winnt\system32\txfaux.dll
es.dll  1999.9.3422.21 231.77 KB (237,328 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\es.dll
loadperf.dll   5.00.2195.1    60.77 KB (62,224 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\loadperf.dll
iasperf.dll    5.00.2160.1    20.27 KB (20,752 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\iasperf.dll
rtipxmib.dll   5.00.2168.1    29.77 KB (30,480 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rtipxmib.dll
```

```
btpagnt.dll    5.00.2195.1284 13.27 KB (13,584 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\btpagnt.dll
ospfagnt.dll   5.00.2195.1284 6.77 KB (6,928 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\ospfagnt.dll
perfos.dll     5.00.2155.1    21.27 KB (21,776 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\perfos.dll
ripagnt.dll    5.00.2195.1284 24.27 KB (24,848 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ripagnt.dll
mcastmib.dll   5.00.2195.1284 13.27 KB (13,584 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mcastmib.dll
igmpagnt.dll   5.00.2195.1284 8.77 KB (8,976 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\igmpagnt.dll
acsmib.dll     5.00.2167.1    11.27 KB (11,536 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\acsmib.dll
evntagnt.dll   5.00.2167.1    94.77 KB (97,040 bytes)       10/4/2000
3:06:01 PM     Microsoft Corporation  c:\winnt\system32\evntagnt.dll
snmpmib.dll    5.00.2134.1    5.77 KB (5,904 bytes)  10/4/2000 3:06:00 PM
       Microsoft Corporation  c:\winnt\system32\snmpmib.dll
inetmib1.dll   5.00.2195.1284 28.77 KB (29,456 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\inetmib1.dll
lmmib2.dll     5.00.2134.1    29.27 KB (29,968 bytes)       10/4/2000
3:06:00 PM     Microsoft Corporation  c:\winnt\system32\lmmib2.dll
snmpapi.dll    5.00.2134.1    17.27 KB (17,680 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\snmpapi.dll
snmp.exe       5.00.2173.1    29.77 KB (30,480 bytes)       10/4/2000
3:06:01 PM     Microsoft Corporation  c:\winnt\system32\snmp.exe
wshnetbs.dll   5.00.2134.1    7.77 KB (7,952 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\wshnetbs.dll
rasadhlp.dll   5.00.2168.1    7.27 KB (7,440 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\rasadhlp.dll
winrnr.dll     5.00.2195.1175 19.27 KB (19,728 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\winrnr.dll
rnr20.dll      5.00.2195.1207 35.77 KB (36,624 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rnr20.dll
rpcss.dll      5.00.2195.1600 229.27 KB (234,768 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rpcss.dll
svchost.exe    5.00.2134.1    7.77 KB (7,952 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\svchost.exe
procconmd8.exe 1, 0, 1, 23    24.27 KB (24,848 bytes)       10/3/2000
8:36:28 AM     Sequent Computer Systems, Inc.
       c:\winnt\system32\procconmd8.exe
regsvc.exe     5.00.2195.31   65.27 KB (66,832 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\regsvc.exe
ntmarta.dll    5.00.2158.1    98.77 KB (101,136 bytes)      8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntmarta.dll
psapi.dll      5.00.2134.1    28.27 KB (28,944 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\psapi.dll
msvcp50.dll    5.00.7051      552.50 KB (565,760 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msvcp50.dll
procconsvc.exe 1, 0, 1, 23    72.27 KB (74,000 bytes)       10/3/2000
8:36:28 AM     Sequent Computer Systems, Inc.
       c:\winnt\system32\procconsvc.exe
vipl.dll       4.01.00 80.00 KB (81,920 bytes)       10/4/2000 9:59:37 AM
       Giganet Incorporated   c:\winnt\system32\vipl.dll
gnconmgr.exe   4.01.00 128.05 KB (131,119 bytes)     8/16/1999 7:09:37 AM
       Giganet Incorporated   c:\winnt\system32\gnconmgr.exe
wshtcpip.dll   5.00.2134.1    17.27 KB (17,680 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wshtcpip.dll
dhcpcsvc.dll   5.00.2195.1600 85.27 KB (87,312 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\dhcpcsvc.dll
tapi32.dll     5.00.2195.1600 124.27 KB (127,248 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\tapi32.dll
rasman.dll     5.00.2188.1    54.77 KB (56,080 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasman.dll
rasapi32.dll   5.00.2188.1    187.77 KB (192,272 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasapi32.dll
icmp.dll       5.00.2134.1    7.27 KB (7,440 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\icmp.dll
iphlpapi.dll   5.00.2173.2    67.77 KB (69,392 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\iphlpapi.dll
msafd.dll      5.00.2195.1614 102.77 KB (105,232 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msafd.dll
scecli.dll     5.00.2191.1    105.27 KB (107,792 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\scecli.dll
atl.dll 3.00.8449      57.56 KB (58,938 bytes)       8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\atl.dll
certcli.dll    5.00.2175.1    132.27 KB (135,440 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\certcli.dll
esent.dll      6.0.3940.4     1.08 MB (1,135,888 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\esent.dll
mswsock.dll    5.00.2195.1207 62.77 KB (64,272 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mswsock.dll
ntdsatq.dll    5.00.2195.1284 31.27 KB (32,016 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntdsatq.dll
ntdsa.dll      5.00.2195.1600 987.27 KB (1,010,960 bytes)   8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntdsa.dll
kdcsvc.dll     5.00.2195.1284 133.77 KB (136,976 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\kdcsvc.dll
sfmapi.dll     5.00.2134.1    38.77 KB (39,696 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\sfmapi.dll
rtutils.dll    5.00.2168.1    43.77 KB (44,816 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rtutils.dll
adsldpc.dll    5.00.2195.1600 125.77 KB (128,784 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\adsldpc.dll
activeds.dll   5.00.2172.1    172.77 KB (176,912 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\activeds.dll
mprapi.dll     5.00.2181.1    79.27 KB (81,168 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mprapi.dll
rassfm.dll     5.00.2195.1179 21.27 KB (21,776 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rassfm.dll
schannel.dll   5.00.2195.1163 137.27 KB (140,560 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\schannel.dll
netlogon.dll   5.00.2195.1600 348.27 KB (356,624 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netlogon.dll
msv1_0.dll     5.00.2195.1620 92.77 KB (94,992 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msv1_0.dll
kerberos.dll   5.00.2195.1378 197.77 KB (202,512 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\kerberos.dll
msprivs.dll    5.00.2195.1600 41.50 KB (42,496 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msprivs.dll
samsrv.dll     5.00.2195.1609 343.27 KB (351,504 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\samsrv.dll
lsasrv.dll     5.00.2195.1620 475.27 KB (486,672 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\lsasrv.dll
lsass.exe      5.00.2195.1620 32.77 KB (33,552 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\lsass.exe
```

```
wmicore.dll    5.00.2178.1    70.77 KB (72,464 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wmicore.dll
psbase.dll     5.00.2195.1600 111.77 KB (114,448 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\psbase.dll
cryptsvc.dll   5.00.2181.1    61.77 KB (63,248 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\cryptsvc.dll
cryptdll.dll   5.00.2135.1    43.27 KB (44,304 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\cryptdll.dll
wkssvc.dll     5.00.2195.1175 95.27 KB (97,552 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wkssvc.dll
srvsvc.dll     5.00.2178.1    79.27 KB (81,168 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\srvsvc.dll
cfgmgr32.dll   5.00.2195.1608 16.77 KB (17,168 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\cfgmgr32.dll
dmserver.dll   2195.23.297.2  11.77 KB (12,048 bytes)       8/8/2000
5:00:00 AM     VERITAS Software Corp. c:\winnt\system32\dmserver.dll
winsta.dll     5.00.2195.32   36.27 KB (37,136 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\winsta.dll
eventlog.dll   5.00.2178.1    43.77 KB (44,816 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\eventlog.dll
ntdsapi.dll    5.00.2195.1175 56.27 KB (57,616 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntdsapi.dll
scesrv.dll     5.00.2188.1    225.77 KB (231,184 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\scesrv.dll
umpnpmgr.dll   5.00.2182.1    86.27 KB (88,336 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\umpnpmgr.dll
services.exe   5.00.2134.1    86.77 KB (88,848 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\services.exe
clbcatq.dll    2000.2.3449.0  496.27 KB (508,176 bytes)     10/3/2000
8:36:03 AM     Microsoft Corporation  c:\winnt\system32\clbcatq.dll
oleaut32.dll   2.40.4514      600.27 KB (614,672 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\oleaut32.dll
netmsg.dll     5.00.2137.1    152.50 KB (156,160 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netmsg.dll
comdlg32.dll   5.00.3103.1000 236.77 KB (242,448 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\comdlg32.dll
netui2.dll     5.00.2134.1    280.27 KB (286,992 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netui2.dll
mprui.dll      5.00.2134.1    54.77 KB (56,080 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\mprui.dll
netui1.dll     5.00.2134.1    210.27 KB (215,312 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netui1.dll
netui0.dll     5.00.2134.1    70.27 KB (71,952 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netui0.dll
ntlanman.dll   5.00.2157.1    35.27 KB (36,112 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntlanman.dll
mpr.dll 5.00.2195.1340 53.27 KB (54,544 bytes)       8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\mpr.dll
cscui.dll      5.00.2195.1387 227.27 KB (232,720 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\cscui.dll
winspool.drv   5.00.2195.1340 109.77 KB (112,400 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\winspool.drv
winscard.dll   5.00.2134.1    77.27 KB (79,120 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\winscard.dll
wlnotify.dll   5.00.2195.1163 53.27 KB (54,544 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wlnotify.dll
cscdll.dll     5.00.2195.1600 98.27 KB (100,624 bytes)      8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\cscdll.dll
```

```
lz32.dll       5.00.2134.1    9.77 KB (10,000 bytes) 8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\lz32.dll
version.dll    5.00.2134.1    15.77 KB (16,144 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\version.dll
rsabase.dll    5.00.2195.1391 129.27 KB (132,368 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rsabase.dll
mscat32.dll    5.131.2134.1   7.77 KB (7,952 bytes)  8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\mscat32.dll
ole32.dll      5.00.2195.1607 965.27 KB (988,432 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ole32.dll
imagehlp.dll   5.00.2195.1620 120.77 KB (123,664 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\imagehlp.dll
msasn1.dll     5.00.2134.1    51.27 KB (52,496 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msasn1.dll
crypt32.dll    5.131.2195.1340        464.77 KB (475,920 bytes)
       8/8/2000 5:00:00 AM    Microsoft Corporation
       c:\winnt\system32\crypt32.dll
wintrust.dll   5.131.2143.1   162.27 KB (166,160 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wintrust.dll
setupapi.dll   5.00.2195.1608 552.77 KB (566,032 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\setupapi.dll
winmm.dll      5.00.2161.1    184.77 KB (189,200 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\winmm.dll
comctl32.dll   5.81    537.77 KB (550,672 bytes)     8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\comctl32.dll
shlwapi.dll    5.00.3103.1000 282.27 KB (289,040 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\shlwapi.dll
shell32.dll    5.00.3103.1000 2.25 MB (2,358,032 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\shell32.dll
msgina.dll     5.00.2195.1600 323.27 KB (331,024 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msgina.dll
wsock32.dll    5.00.2195.1207 21.27 KB (21,776 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wsock32.dll
dnsapi.dll     5.00.2195.1600 127.77 KB (130,832 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\dnsapi.dll
wldap32.dll    5.00.2195.1175 155.27 KB (158,992 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\wldap32.dll
ws2help.dll    5.00.2134.1    17.77 KB (18,192 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ws2help.dll
ws2_32.dll     5.00.2195.1340 68.77 KB (70,416 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ws2_32.dll
samlib.dll     5.00.2195.1175 46.27 KB (47,376 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\samlib.dll
netrap.dll     5.00.2134.1    11.27 KB (11,536 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netrap.dll
netapi32.dll   5.00.2195.1600 303.27 KB (310,544 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\netapi32.dll
profmap.dll    5.00.2181.1    29.27 KB (29,968 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\profmap.dll
secur32.dll    5.00.2195.1600 47.27 KB (48,400 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\secur32.dll
sfc.dll 5.00.2195.1618 85.27 KB (87,312 bytes)       8/8/2000 5:00:00 AM
       Microsoft Corporation  c:\winnt\system32\sfc.dll
nddeapi.dll    5.00.2137.1    15.27 KB (15,632 bytes)       8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\nddeapi.dll
userenv.dll    5.00.2195.1600 359.27 KB (367,888 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\userenv.dll
user32.dll     5.00.2195.1600 392.77 KB (402,192 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\user32.dll
```

```
gdi32.dll      5.00.2195.1340 228.77 KB (234,256 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\gdi32.dll
rpcrt4.dll     5.00.2195.1615 436.27 KB (446,736 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\rpcrt4.dll
advapi32.dll   5.00.2195.1600 349.27 KB (357,648 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\advapi32.dll
kernel32.dll   5.00.2195.1600 713.27 KB (730,384 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\kernel32.dll
msvcrt.dll     6.10.8637.0    288.09 KB (295,000 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\msvcrt.dll
winlogon.exe   5.00.2195.1600 172.77 KB (176,912 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\winlogon.exe
sfcfiles.dll   5.00.2195.1620 885.27 KB (906,512 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\sfcfiles.dll
ntdll.dll      5.00.2195.1600 475.27 KB (486,672 bytes)     8/8/2000
5:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntdll.dll
smss.exe       5.00.2195.31   44.27 KB (45,328 bytes)      8/8/2000
5:00:00 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 Manual  Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Application Management AppMgmt Stopped Manual  Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Computer Browser       Browser Stopped Manual  Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Indexing Service       cisvc   Stopped Manual  Share Process
       c:\winnt\system32\cisvc.exe   Normal  LocalSystem    0
ClipBook       ClipSrv Stopped Manual  Own Process
       c:\winnt\system32\clipsrv.exe Normal  LocalSystem    0
Distributed File System        Dfs     Stopped Disabled       Own Process
       c:\winnt\system32\dfssvc.exe  Normal  LocalSystem    0
DHCP Client    Dhcp    Stopped Disabled       Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Logical Disk Manager Administrative Service  dmadmin Stopped Manual  Share
Process c:\winnt\system32\dmadmin.exe /com    Normal  LocalSystem    0
Logical Disk Manager   dmserver       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
Mylex Global Array Manager Server     gamscm  Stopped Manual  Own Process
       c:\winnt\system32\gamserv\gamscm.exe  Normal  LocalSystem    0
cLAN Connection Manager        GniConMgr       Running Auto    Own Process
       c:\winnt\system32\gnconmgr.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 Manual  Own Process
       c:\winnt\system32\llssrv.exe  Normal  LocalSystem    0
TCP/IP NetBIOS Helper Service LmHosts Stopped Manual  Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Messenger      Messenger       Stopped Manual  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   Stopped Manual  Own Process
       c:\winnt\system32\msdtc.exe   Normal  LocalSystem    0
Windows Installer      MSIServer       Stopped Manual  Share Process
       c:\winnt\system32\msiexec.exe /v      Normal  LocalSystem    0
MSSQLSERVER    MSSQLSERVER     Stopped Manual  Own Process
       c:\progra~1\micros~2\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
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
Process Control Service        ProcCon Running Auto    Own Process
       c:\winnt\system32\procconsvc.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  Running 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       Stopped Manual  Share Process
       c:\winnt\system32\mstask.exe  Normal  LocalSystem    0
RunAs Service  seclogon       Stopped Manual  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
SNMP Service   SNMP    Running Auto    Own Process
       c:\winnt\system32\snmp.exe    Normal  LocalSystem    0
SNMP Trap Service      SNMPTRAP       Stopped Manual  Own Process
       c:\winnt\system32\snmptrap.exe        Normal  LocalSystem    0
Print Spooler  Spooler Stopped Disabled       Own Process
       c:\winnt\system32\spoolsv.exe Normal  LocalSystem    0
SQLSERVERAGENT SQLSERVERAGENT Stopped Manual  Own Process
       c:\progra~1\micros~2\mssql\binn\sqlagent.exe Normal  LocalSystem
       0
Performance Logs and Alerts   SysmonLog      Stopped Manual  Own Process
       c:\winnt\system32\smlogsvc.exe        Normal  LocalSystem    0
Telephony      TapiSrv Running Manual  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 Manual  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
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
Adobe Acrobat  All Users:Adobe Acrobat       All Users
GigaNet All Users:GigaNet      All Users
Microsoft SQL Server   All Users:Microsoft SQL Server       All Users
Startup All Users:Startup      All Users
WinZip  All Users:WinZip       All Users
Accessories    CAPRICORN8\Administrator:Accessories
       CAPRICORN8\Administrator
Accessories\Accessibility
       CAPRICORN8\Administrator:Accessories\Accessibility
       CAPRICORN8\Administrator
Accessories\Entertainment
       CAPRICORN8\Administrator:Accessories\Entertainment
       CAPRICORN8\Administrator
Accessories\System Tools      CAPRICORN8\Administrator:Accessories\System
Tools   CAPRICORN8\Administrator
Administrative Tools   CAPRICORN8\Administrator:Administrative Tools
       CAPRICORN8\Administrator
Startup CAPRICORN8\Administrator:Startup      CAPRICORN8\Administrator

[Startup Programs]

Program Command User Name      Location
No startup program information

[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   C:\WINNT\System32\mspaint.exe
Adobe Acrobat Document C:\Acrobat3\Reader\AcroRd32.exe

[Internet Explorer 5]

[ Following are sub-categories of this main category ]
```

```
[Summary]

Item    Value
Version 5.00.3103.1000
Build   53103.1000
Product ID      53567-OEM-0000007-00000
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    5.0.2195.1600   349 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
advpack.dll     5.0.3103.1000   87 KB   8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
browselc.dll    5.0.3103.1000   35 KB   8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
browseui.dll    5.0.3103.1000   789 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
ckcnv.exe       5.0.2195.1600   9 KB    8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
comctl32.dll    5.81.3103.1000 538 KB   8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
crypt32.dll     5.131.2195.1340         465 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
enhsig.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     5.0.3103.1000   57 KB   8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
iexplore.exe    5.0.3103.1000   59 KB   8/8/2000 4:00:00 AM     C:\Program
Files\Internet Explorer         Microsoft Corporation
imagehlp.dll    5.0.2195.1620   121 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
imghelp.dll     <File Missing> Not Available  Not Available  Not Available
        Not Available
inseng.dll      5.0.3103.1000   72 KB   8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
jobexec.dll     5.0.0.1 47 KB   8/8/2000 4:00:00 AM     C:\WINNT\system32
        Microsoft Corporation
jscript.dll     5.1.0.5010      476 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
jsproxy.dll     5.0.3103.1000   13 KB   8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
msaahtml.dll    <File Missing> Not Available  Not Available  Not Available
        Not Available
mshtml.dll      5.0.3103.1000   2292 KB 8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
msjava.dll      5.0.3310.0      922 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
msoss.dll       <File Missing> Not Available  Not Available  Not Available
        Not Available
msxml.dll       8.0.5226.0      506 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
occache.dll     5.0.3103.1000   86 KB   8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
ole32.dll       5.0.2195.1607   965 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
oleaut32.dll    2.40.4514.1     600 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
olepro32.dll    5.0.4514.1      160 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
rsabase.dll     5.0.2195.1391   129 KB  8/8/2000 4:00:00 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    5.0.2195.0      137 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
shdoc401.dll    <File Missing> Not Available  Not Available  Not Available
        Not Available
shdocvw.dll     5.0.3103.1000   1078 KB 8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
shell32.dll     5.0.3103.1000   2303 KB 8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
shlwapi.dll     5.0.3103.1000   282 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
url.dll 5.0.3103.1000   82 KB   8/8/2000 4:00:00 AM     C:\WINNT\system32
        Microsoft Corporation
urlmon.dll      5.0.3103.1000   441 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
vbscript.dll    5.1.0.5010      428 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
webcheck.dll    5.0.3103.1000   252 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
win.com 5.0.2134.1      24 KB   8/8/2000 4:00:00 AM     C:\WINNT\system32
        Microsoft Corporation
wininet.dll     5.0.3103.1000   457 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
winsock.dll     3.10.0.103      3 KB    8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
wintrust.dll    5.131.2143.1    162 KB  8/8/2000 4:00:00 AM
        C:\WINNT\system32       Microsoft Corporation
wsock.vxd       <File Missing> Not Available  Not Available  Not Available
        Not Available
wsock32.dll     5.0.2195.1207   21 KB   8/8/2000 4:00:00 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        wininet.dll
AutoProxyDetectMode    Disabled
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       17414 MB
Available Disk Space   14293 MB
Maximum Cache Size     544 MB
Available Cache Size   483 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  10/3/2000 to 9/9/2100  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
```

Windows 2000 Datacenter Server Registry Information

Software\GigaNet

```
Key Name:          SOFTWARE\Giganet
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM

Key Name:          SOFTWARE\Giganet\cLAN Management Console
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:59 AM

Key Name:          SOFTWARE\Giganet\cLAN Management Console\1.00.000
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:59 AM

Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM

Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM

Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\BeginModify
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            <NO NAME>
  Type:            REG_SZ
  Data:            07:21:55


Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\Fabric
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM

Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\Fabric\Switch1
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            Ports
  Type:            REG_SZ
  Data:            n1 n2 n3 n4 x x x x

Value 1
  Name:            SwitchModel
  Type:            REG_DWORD
  Data:            0x1388

Value 2
  Name:            SwitchRev
  Type:            REG_DWORD
```

```
  Data:            0x2103


Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\LastModified
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            <NO NAME>
  Type:            REG_SZ
  Data:            07:21:55


Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\NodeList
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM

Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\NodeList\Node0
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            Address
  Type:            REG_SZ
  Data:            0090FA000886

Value 1
  Name:            MediaType
  Type:            REG_SZ
  Data:

Value 2
  Name:            Name
  Type:            REG_SZ
  Data:            CAPRICORN8

Value 3
  Name:            NICName
  Type:            REG_SZ
  Data:            gnivia0

Value 4
  Name:            NodeNumber
  Type:            REG_DWORD
  Data:            0x1

Value 5
  Name:            State
  Type:            REG_DWORD
  Data:            0x1

Value 6
  Name:            SwitchPort
  Type:            REG_SZ
  Data:            s1.1


Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\NodeList\Node2
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            Address
  Type:            REG_SZ
  Data:            0090FA001420

Value 1
  Name:            MediaType
  Type:            REG_SZ
  Data:

Value 2
  Name:            Name
  Type:            REG_SZ
  Data:            CLIENT2

Value 3
  Name:            NICName
  Type:            REG_SZ
  Data:            nic0

Value 4
  Name:            NodeNumber
  Type:            REG_DWORD
  Data:            0x2

Value 5
  Name:            State
  Type:            REG_DWORD
  Data:            0x1

Value 6
  Name:            SwitchPort
  Type:            REG_SZ
  Data:            s1.2


Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\NodeList\Node3
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            Address
  Type:            REG_SZ
  Data:            0090FA00136E

Value 1
  Name:            MediaType
  Type:            REG_SZ
  Data:

Value 2
  Name:            Name
  Type:            REG_SZ
  Data:            CLIENT3

Value 3
  Name:            NICName
  Type:            REG_SZ
  Data:            nic0

Value 4
```

```
  Name:            NodeNumber
  Type:            REG_DWORD
  Data:            0x3

Value 5
  Name:            State
  Type:            REG_DWORD
  Data:            0x1

Value 6
  Name:            SwitchPort
  Type:            REG_SZ
  Data:            s1.3


Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\0\NodeList\Node4
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            Address
  Type:            REG_SZ
  Data:            0090FA0008FA

Value 1
  Name:            MediaType
  Type:            REG_SZ
  Data:

Value 2
  Name:            Name
  Type:            REG_SZ
  Data:            CLIENT1

Value 3
  Name:            NICName
  Type:            REG_SZ
  Data:            nic0

Value 4
  Name:            NodeNumber
  Type:            REG_DWORD
  Data:            0x4

Value 5
  Name:            State
  Type:            REG_DWORD
  Data:            0x1

Value 6
  Name:            SwitchPort
  Type:            REG_SZ
  Data:            s1.4


Key Name:          SOFTWARE\Giganet\GniConMgrMgmt2\LastModified
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:21 AM
Value 0
  Name:            <NO NAME>
  Type:            REG_SZ
  Data:            07:21:55
```

Software\Microsoft\MSSQLServer

```
Key Name:          SOFTWARE\Microsoft\MSSQLServer
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:35 PM

Key Name:          SOFTWARE\Microsoft\MSSQLServer\Client
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            SharedMemoryOn
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Client\ConnectTo
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            DSQUERY
  Type:            REG_SZ
  Data:            DBNETLIB


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Client\DB-Lib
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AutoAnsiToOem
  Type:            REG_SZ
  Data:            ON

Value 1
  Name:            UseIntlSettings
  Type:            REG_SZ
  Data:            ON


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Client\SuperSocketNetLib
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:41 PM
Value 0
  Name:            Encrypt
  Type:            REG_DWORD
  Data:            0

Value 1
  Name:            ProtocolOrder
  Type:            REG_MULTI_SZ
  Data:            via
                   tcp
                   np
```

```
Key Name:
SOFTWARE\Microsoft\MSSQLServer\Client\SuperSocketNetLib\Np
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            DefaultPipe
  Type:            REG_SZ
  Data:            sql\query


Key Name:
SOFTWARE\Microsoft\MSSQLServer\Client\SuperSocketNetLib\Tcp
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            DefaultPort
  Type:            REG_DWORD
  Data:            0x599


Key Name:
SOFTWARE\Microsoft\MSSQLServer\Client\SuperSocketNetLib\VIA
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:41 PM
Value 0
  Name:            DefaultClientNIC
  Type:            REG_SZ
  Data:            0

Value 1
  Name:            DefaultServerPort
  Type:            REG_SZ
  Data:            0:1433

Value 2
  Name:            RecognizedVendors
  Type:            REG_SZ
  Data:            Giganet, ServerNet II

Value 3
  Name:            Vendor
  Type:            REG_SZ
  Data:            Giganet


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Client\TDS
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 4:51 PM
Value 0
  Name:            <NO NAME>
  Type:            REG_SZ
  Data:            7.0

Value 1
  Name:            CAPRICORN8
  Type:            REG_SZ
  Data:            7.0


Key Name:          SOFTWARE\Microsoft\MSSQLServer\MSSQLServer
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:39 PM
Value 0
  Name:            AuditLevel
  Type:            REG_DWORD
  Data:            0

Value 1
  Name:            BackupDirectory
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL Server\MSSQL\BACKUP

Value 2
  Name:            DefaultCollationName
  Type:            REG_SZ
  Data:            Latin1_General_BIN

Value 3
  Name:            DefaultDomain
  Type:            REG_SZ
  Data:            CAPRICORN8

Value 4
  Name:            DefaultLogin
  Type:            REG_SZ
  Data:            guest

Value 5
  Name:            ListenOn
  Type:            REG_MULTI_SZ
  Data:            SSMSSH70
                   SSNETLIB


Value 6
  Name:            LoginMode
  Type:            REG_DWORD
  Data:            0x2

Value 7
  Name:            Map#
  Type:            REG_SZ
  Data:            -

Value 8
  Name:            Map$
  Type:            REG_SZ
  Data:

Value 9
  Name:            Map_
  Type:            REG_SZ
  Data:            \

Value 10
  Name:            ResourceMgrID
  Type:            REG_SZ
```

```
  Data:            {AFAD74A8-2FA5-4915-BCBC-9EAF1F51A8DE}

Value 11
  Name:            SetHostName
  Type:            REG_DWORD
  Data:            0

Value 12
  Name:            Tapeloadwaittime
  Type:            REG_DWORD
  Data:            0xffffffff


Key Name:
SOFTWARE\Microsoft\MSSQLServer\MSSQLServer\CurrentVersion
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            checksum
  Type:            REG_BINARY
  Data:
00000000   37 39 32 32 63 31 35 38 - 61 65 37 64 34 63 64 37
7922c158ae7d4cd7
00000010   35 30 64 61 30 33 34 62 - 37 64 63 33 37 39 35 39
50da034b7dc37959
00000020   36 39 30 37 30 63 64 33 - 30 33 63 65 37 37 31 30
69070cd303ce7710
00000030   64 38 34 31 34 31 34 38 - 64 35 62 63 62 65 32 32
d8414148d5bcbe22
00000040   37 61 39 30 62 63 34 37 - 61 32 35 36 37 37 39 38
7a90bc47a2567798
00000050   33 35 64 64 39 63 34 37 - 33 62 32 33 31 66 64 31
35dd9c473b231fd1
00000060   33 66 30 38 33 35 63 63 - 34 36 66 35 64 39 39 63
3f0835cc46f5d99c
00000070   34 63 39 32 36 36 35 39 - 35 63 64 64 34 34 30 63
4c9266595cdd440c
00000080   33 61 35 66 34 64 36 34 - 30 30 36 39 66 30 39 62
3a5f4d640069f09b
00000090   39 61 63 31 35 62 37 64 - 34 66 63 39 34 33 65 64
9ac15b7d4fc943ed
000000a0   00                                                .

Value 1
  Name:            CurrentVersion
  Type:            REG_SZ
  Data:            8.00.194

Value 2
  Name:            Language
  Type:            REG_DWORD
  Data:            0x409

Value 3
  Name:            RegisteredOwner
  Type:            REG_SZ
  Data:            SAM&M

Value 4
  Name:            SerialNumber
  Type:            REG_DWORD
  Data:            0x81530040


Key Name:          SOFTWARE\Microsoft\MSSQLServer\MSSQLServer\Parameters
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            SQLArg0
  Type:            REG_SZ
  Data:            -dC:\Program Files\Microsoft SQL
Server\MSSQL\data\master.mdf

Value 1
  Name:            SQLArg1
  Type:            REG_SZ
  Data:            -eC:\Program Files\Microsoft SQL
Server\MSSQL\log\ERRORLOG

Value 2
  Name:            SQLArg2
  Type:            REG_SZ
  Data:            -lC:\Program Files\Microsoft SQL
Server\MSSQL\data\mastlog.ldf


Key Name:          SOFTWARE\Microsoft\MSSQLServer\MSSQLServer\RPCNetLib
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            Security
  Type:            REG_SZ
  Data:


Key Name:
SOFTWARE\Microsoft\MSSQLServer\MSSQLServer\SuperSocketNetLib
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:40 PM
Value 0
  Name:            Encrypt
  Type:            REG_DWORD
  Data:            0

Value 1
  Name:            ProtocolList
  Type:            REG_MULTI_SZ
  Data:            np
                   tcp
                   via



Key Name:
SOFTWARE\Microsoft\MSSQLServer\MSSQLServer\SuperSocketNetLib\Np
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
```

```
  Name:            PipeName
  Type:            REG_SZ
  Data:            \\.\pipe\sql\query


Key Name:
SOFTWARE\Microsoft\MSSQLServer\MSSQLServer\SuperSocketNetLib\Tcp
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            TcpDynamicPorts
  Type:            REG_SZ
  Data:

Value 1
  Name:            TcpHideFlag
  Type:            REG_DWORD
  Data:            0

Value 2
  Name:            TcpPort
  Type:            REG_SZ
  Data:            1433


Key Name:
SOFTWARE\Microsoft\MSSQLServer\MSSQLServer\SuperSocketNetLib\VIA
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:40 PM
Value 0
  Name:            ListenInfo
  Type:            REG_SZ
  Data:            0:1433

Value 1
  Name:            RecognizedVendors
  Type:            REG_MULTI_SZ
  Data:            Giganet


Value 2
  Name:            Vendor
  Type:            REG_SZ
  Data:            Giganet


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\ADSDSOObject
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\DB2OLEDB
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:
SOFTWARE\Microsoft\MSSQLServer\Providers\Microsoft.Jet.OLEDB.4.0
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\MSDAORA
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\MSDASQL
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\MSIDXS
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\MSQLImpProv
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1
```

```
Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\MSSEARCHSQL
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Providers\SQLOLEDB
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            AllowInProcess
  Type:            REG_DWORD
  Data:            0x1

Value 1
  Name:            DisallowAdhocAccess
  Type:            REG_DWORD
  Data:            0


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Setup
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:39 PM
Value 0
  Name:            firststart
  Type:            REG_DWORD
  Data:            0

Value 1
  Name:            SourcePath
  Type:            REG_SZ
  Data:            \\mv-nas-4\shared\SYSTEM~1\MICROS~1\SQL_2K~4

Value 2
  Name:            SQLDataRoot
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL Server\MSSQL

Value 3
  Name:            SQLPath
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL Server\MSSQL


Key Name:          SOFTWARE\Microsoft\MSSQLServer\SQLServerAgent
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            DownloadedMaxRows
  Type:            REG_DWORD
  Data:            0x64

Value 1
  Name:            ErrorLogFile
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\LOG\SQLAGENT.OUT

Value 2
  Name:            ErrorLoggingLevel
  Type:            REG_DWORD
  Data:            0x3

Value 3
  Name:            JobHistoryMaxRows
  Type:            REG_DWORD
  Data:            0x3e8

Value 4
  Name:            JobHistoryMaxRowsPerJob
  Type:            REG_DWORD
  Data:            0x64

Value 5
  Name:            MSXServerName
  Type:            REG_SZ
  Data:

Value 6
  Name:            NonAlertableErrors
  Type:            REG_SZ
  Data:            1204,4002

Value 7
  Name:            RestartServer
  Type:            REG_DWORD
  Data:            0x1

Value 8
  Name:            ServerHost
  Type:            REG_SZ
  Data:

Value 9
  Name:            SysAdminOnly
  Type:            REG_DWORD
  Data:            0x1

Value 10
  Name:            WorkingDirectory
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL Server\MSSQL\JOBS


Key Name:
SOFTWARE\Microsoft\MSSQLServer\SQLServerAgent\Subsystems
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            ActiveScripting
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\BINN\SQLATXSS.DLL,NULL,ActiveScriptStart,ActiveScriptEvent,Ac
tiveScriptStop,10
```

```
Value 1
  Name:            CmdExec
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\BINN\SQLCMDSS.DLL,NULL,CmdExecStart,CmdEvent,CmdExecStop,10

Value 2
  Name:            Distribution
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\BINN\SQLREPSS.DLL,C:\Program Files\Microsoft SQL
Server\80\COM\DISTRIB.EXE,ReplStart,ReplEvent,ReplStop,100

Value 3
  Name:            LogReader
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\BINN\SQLREPSS.DLL,C:\Program Files\Microsoft SQL
Server\80\COM\LOGREAD.EXE,ReplStart,ReplEvent,ReplStop,25

Value 4
  Name:            Merge
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\BINN\SQLREPSS.DLL,C:\Program Files\Microsoft SQL
Server\80\COM\REPLMERG.EXE,ReplStart,ReplEvent,ReplStop,100

Value 5
  Name:            QueueReader
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\BINN\SQLREPSS.DLL,C:\Program Files\Microsoft SQL
Server\80\COM\QRDRSVC.EXE,ReplStart,ReplEvent,ReplStop,100

Value 6
  Name:            Snapshot
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\MSSQL\BINN\SQLREPSS.DLL,C:\Program Files\Microsoft SQL
Server\80\COM\SNAPSHOT.EXE,ReplStart,ReplEvent,ReplStop,100


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Tracking
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            {6DC86044-0C71-11d3-9E18-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 1
  Name:            {E07FDDA4-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 2
  Name:            {E07FDDA8-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 3
  Name:            {E07FDDA9-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 4
  Name:            {E07FDDAA-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 5
  Name:            {E07FDDAB-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 6
  Name:            {E07FDDAC-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 7
  Name:            {E07FDDAD-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 8
  Name:            {E07FDDAF-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 9
  Name:            {E07FDDB2-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 10
  Name:            {E07FDDBE-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 11
  Name:            {E07FDDBF-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:

Value 12
  Name:            {E07FDDC0-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_SZ
  Data:


Key Name:          SOFTWARE\Microsoft\MSSQLServer\Tracking\Shortcuts
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            Books Online
  Type:            REG_SZ
```

```
  Data:

Value 1
  Name:            Client Network Utility
  Type:            REG_SZ
  Data:

Value 2
  Name:            Configure SQL XML Support in IIS
  Type:            REG_SZ
  Data:

Value 3
  Name:            Enterprise Manager
  Type:            REG_SZ
  Data:

Value 4
  Name:            Import and Export Data
  Type:            REG_SZ
  Data:

Value 5
  Name:            Profiler
  Type:            REG_SZ
  Data:

Value 6
  Name:            Query Analyzer
  Type:            REG_SZ
  Data:

Value 7
  Name:            Server Network Utility
  Type:            REG_SZ
  Data:

Value 8
  Name:            Service Manager
  Type:            REG_SZ
  Data:
```

Software\Microsoft\Microsoft SQL Server

```
Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:23 AM
Value 0
  Name:            InstalledInstances
  Type:            REG_MULTI_SZ
  Data:            MSSQLSERVER


Value 1
  Name:            SqlMdacRegRefCount
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            SsrpActiveServer
  Type:            REG_SZ
  Data:            MSSQLServer


Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server\8.00.000
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:34 PM

Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server\80
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:34 PM

Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\MSSQLLicenseInfo
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:34 PM

Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\MSSQLLicenseInfo\MSSQL8.00
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:34 PM
Value 0
  Name:            ConcurrentLimit
  Type:            REG_DWORD
  Data:            0x8

Value 1
  Name:            DisplayName
  Type:            REG_SZ
  Data:            SQL Server 2000

Value 2
  Name:            FamilyDisplayName
  Type:            REG_SZ
  Data:            Microsoft SQL Server

Value 3
  Name:            FlipAllow
  Type:            REG_DWORD
  Data:            0

Value 4
  Name:            Mode
  Type:            REG_DWORD
  Data:            0x2


Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server\80\Registration
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:34 PM
Value 0
  Name:            CD_KEY
  Type:            REG_SZ
  Data:            BC6JV-WVYQ9-PC8H6-W938R-8BWJD
```

```
Value 1
  Name:            DigitalProductID
  Type:            REG_BINARY
  Data:
00000000   a4 00 00 00 03 00 00 00 - 35 33 39 33 34 2d 30 30
¤.......53934-00
00000010   30 2d 30 30 30 30 30 39 - 35 2d 30 35 39 39 38 00  0-0000095-
05998.
00000020   0a 00 00 00 38 31 30 2d - 30 30 35 36 30 00 00 00  ....810-
00560...
00000030   00 00 00 00 12 00 00 00 - 00 a6 0a 0f 5e 9d 9e ef
.........¦..^..ï
00000040   0d 05 00 00 00 00 00 00 - 9f 3f db 39 c0 12 13 00
.........?Û9À...
00000050   00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
................
00000060   00 00 00 00 00 00 00 00 - 35 37 35 32 35 00 00 00
........57525...
00000070   00 00 00 00 e3 0c 00 00 - e1 80 41 e4 00 08 00 00
....ã...á.Aä....
00000080   a4 15 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
¤...............
00000090   00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00
................
000000a0   45 ee dd ac                                       EîÝ¬

Value 2
  Name:            ProductID
  Type:            REG_SZ
  Data:            53934-000-0000095-05998


Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server\80\Replication
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM

Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Replication\MergeReplicationProvider
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM

Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Replication\MergeReplicationProvider\MsJet
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            <NO NAME>
  Type:            REG_SZ
  Data:            {f159cf30-0db4-11d1-b272-00aa00b8de95}


Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server\80\Tools
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM

Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Tools\ClientSetup
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            SQLPath
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL Server\80\Tools


Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Tools\ClientSetup\CurrentVersion
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:37 PM
Value 0
  Name:            CurrentVersion
  Type:            REG_SZ
  Data:            8.00.194

Value 1
  Name:            Language
  Type:            REG_DWORD
  Data:            0x409


Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server\80\Tools\Help
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            HelpFile
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\80\Tools\Books\SQL80.col

Value 1
  Name:            HelpLCID
  Type:            REG_SZ
  Data:            1033

Value 2
  Name:            HelpPath
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL Server\80\Tools\Books

Value 3
  Name:            RefCount
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Tools\Service Manager
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            Action Verify
  Type:            REG_DWORD
  Data:            0

Value 1
  Name:            Default
  Type:            REG_SZ
  Data:            MSSQLSERVER
```

```
Value 2
  Name:            Remote
  Type:            REG_DWORD
  Data:            0x1

Value 3
  Name:            Services
  Type:            REG_MULTI_SZ
  Data:            MSDTC



Key Name:          SOFTWARE\Microsoft\Microsoft SQL Server\80\Tools\SQLEW
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            CustomizableAlertBaseURL
  Type:            REG_SZ
  Data:

Value 1
  Name:            CustomizableAlertDefaultBtnShortText
  Type:            REG_SZ
  Data:            Feedbac&k

Value 2
  Name:            CustomizableAlertDefaultButtonText
  Type:            REG_SZ
  Data:            &Send Feedback


Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Tools\SQLEW\Replication
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            PerfmonFile
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\80\Tools\BINN\REPLMON.PMC


Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Tools\SQLEW\Wizards
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            Web Assistant
  Type:            REG_SZ
  Data:            C:\Program Files\Microsoft SQL
Server\80\Tools\BINN\semwebwz.DLL^WebWizardEntry


Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Tools\Tracking
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            {E07FDDA8-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 1
  Name:            {E07FDDA9-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            {E07FDDAA-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 3
  Name:            {E07FDDAB-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 4
  Name:            {E07FDDAC-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 5
  Name:            {E07FDDAD-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 6
  Name:            {E07FDDAF-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 7
  Name:            {E07FDDB2-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 8
  Name:            {E07FDDBE-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1

Value 9
  Name:            {E07FDDBF-5A21-11d2-9DAD-00C04F79D434}
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SOFTWARE\Microsoft\Microsoft SQL
Server\80\Tools\Tracking\Shortcuts
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            Books Online
  Type:            REG_DWORD
  Data:            0x1
```

```
Value 1
  Name:            Client Network Utility
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            Configure SQL XML Support in IIS
  Type:            REG_DWORD
  Data:            0x1

Value 3
  Name:            Enterprise Manager
  Type:            REG_DWORD
  Data:            0x1

Value 4
  Name:            Import and Export Data
  Type:            REG_DWORD
  Data:            0x1

Value 5
  Name:            Profiler
  Type:            REG_DWORD
  Data:            0x1

Value 6
  Name:            Query Analyzer
  Type:            REG_DWORD
  Data:            0x1

Value 7
  Name:            Server Network Utility
  Type:            REG_DWORD
  Data:            0x1

Value 8
  Name:            Service Manager
  Type:            REG_DWORD
  Data:            0x1
```

Software/Mylex

```
Key Name:          SOFTWARE\Mylex
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:30 PM

Key Name:          SOFTWARE\Mylex\Global Array Manager Client
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:30 PM

Key Name:          SOFTWARE\Mylex\Global Array Manager Client\3.01-00
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:30 PM
Value 0
  Name:            Company Name
  Type:            REG_SZ
  Data:            unisys

Value 1
  Name:            User Name
  Type:            REG_SZ
  Data:            mv-samm


Key Name:          SOFTWARE\Mylex\Global Array Manager Server
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:30 PM

Key Name:          SOFTWARE\Mylex\Global Array Manager Server\3.01-02
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:30 PM
```

CurrentControlSet\Control\Session Manager\I/O System

```
Key Name:          SYSTEM\CurrentControlSet\Control\Session Manager\I/O
System
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:06 AM
Value 0
  Name:            CountOperations
  Type:            REG_DWORD
  Data:            0

Value 1
  Name:            LargeIrpStackLocations
  Type:            REG_DWORD
  Data:            0x7
```

CurrentControlSet\Control\Session Manager\Memory Management

```
Key Name:          SYSTEM\CurrentControlSet\Control\Session Manager\Memory
Management
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:07 AM
Value 0
  Name:            ClearPageFileAtShutdown
  Type:            REG_DWORD
  Data:            0

Value 1
  Name:            DisablePagingExecutive
  Type:            REG_DWORD
  Data:            0

Value 2
  Name:            DontVerifyRandomDrivers
  Type:            REG_DWORD
  Data:            0x1

Value 3
```

```
  Name:            IoPageLockLimit
  Type:            REG_DWORD
  Data:            0

Value 4
  Name:            LargeSystemCache
  Type:            REG_DWORD
  Data:            0

Value 5
  Name:            NonPagedPoolQuota
  Type:            REG_DWORD
  Data:            0

Value 6
  Name:            NonPagedPoolSize
  Type:            REG_DWORD
  Data:            0

Value 7
  Name:            PagedPoolQuota
  Type:            REG_DWORD
  Data:            0

Value 8
  Name:            PagedPoolSize
  Type:            REG_DWORD
  Data:            0

Value 9
  Name:            PagingFiles
  Type:            REG_MULTI_SZ
  Data:            C:\pagefile.sys 2046 4092


Value 10
  Name:            PhysicalAddressExtension
  Type:            REG_DWORD
  Data:            0x1

Value 11
  Name:            SecondLevelDataCache
  Type:            REG_DWORD
  Data:            0

Value 12
  Name:            SystemPages
  Type:            REG_DWORD
  Data:            0
```

CurrentControlSet\Services\dac2w2k

```
Key Name:          SYSTEM\CurrentControlSet\Services\dac2w2k
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:11 PM
Value 0
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 1
  Name:            Group
  Type:            REG_SZ
  Data:            SCSI Miniport

Value 2
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            System32\DRIVERS\dac2w2k.sys

Value 3
  Name:            Start
  Type:            REG_DWORD
  Data:            0

Value 4
  Name:            Tag
  Type:            REG_DWORD
  Data:            0x21

Value 5
  Name:            Type
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\dac2w2k\Enum
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:20 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&3a654c6b&0&4028

Value 1
  Name:            1
  Type:            REG_SZ
  Data:
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&2a769d37&0&4030

Value 2
  Name:            2
  Type:            REG_SZ
  Data:
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&2fbc1dea&0&4038

Value 3
  Name:            3
  Type:            REG_SZ
  Data:
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&8c49857&0&4020

Value 4
  Name:            4
```

```
  Type:            REG_SZ
  Data:
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&375c4928&0&4028

Value 5
  Name:            5
  Type:            REG_SZ
  Data:
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&1b89a02&0&4020

Value 6
  Name:            6
  Type:            REG_SZ
  Data:
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&1cdf5718&0&4028

Value 7
  Name:            Count
  Type:            REG_DWORD
  Data:            0x7

Value 8
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0x7


Key Name:          SYSTEM\CurrentControlSet\Services\dac2w2k\Parameters
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:11 PM

Key Name:
SYSTEM\CurrentControlSet\Services\dac2w2k\Parameters\Device
Class Name:        <NO CLASS>
Last Write Time:   10/27/2000 - 8:51 PM
Value 0
  Name:            DriverParameter
  Type:            REG_SZ
  Data:            ConfigureSIR=13


Key Name:
SYSTEM\CurrentControlSet\Services\dac2w2k\Parameters\PnpInterface
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:11 PM
Value 0
  Name:            5
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\dac2w2k\Security
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:11 PM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
```

```
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 00 00 00 00
................
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 98 3a 00 00 00 00 18 00   ...
....:......
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 98 3a 00 00  ....
...#....:..
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\ Diskperf

```
Key Name:          SYSTEM\CurrentControlSet\Services\Diskperf
Class Name:        <NO CLASS>
Last Write Time:   10/28/2000 - 3:22 PM
Value 0
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 1
  Name:            Group
  Type:            REG_SZ
  Data:            System Bus Extender

Value 2
  Name:            Start
  Type:            REG_DWORD
  Data:            0x4

Value 3
  Name:            Tag
  Type:            REG_DWORD
  Data:            0xa

Value 4
  Name:            Type
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\Diskperf\Enum
```

```
Class Name:         <NO CLASS>
Last Write Time:    10/30/2000 - 7:20 AM
Value 0
  Name:             0
  Type:             REG_SZ
  Data:             Root\LEGACY_DISKPERF\0000

Value 1
  Name:             Count
  Type:             REG_DWORD
  Data:             0x1

Value 2
  Name:             NextInstance
  Type:             REG_DWORD
  Data:             0x1
```

CurrentControlSet\Services\gamdrv

```
Key Name:           SYSTEM\CurrentControlSet\Services\gamdrv
Class Name:         <NO CLASS>
Last Write Time:    10/28/2000 - 7:19 PM
Value 0
  Name:             ErrorControl
  Type:             REG_DWORD
  Data:             0x1

Value 1
  Name:             Group
  Type:             REG_SZ
  Data:             SCSI Class

Value 2
  Name:             ImagePath
  Type:             REG_EXPAND_SZ
  Data:             system32\Drivers\Gamdrv.sys

Value 3
  Name:             Start
  Type:             REG_DWORD
  Data:             0

Value 4
  Name:             Type
  Type:             REG_DWORD
  Data:             0x1


Key Name:           SYSTEM\CurrentControlSet\Services\gamdrv\Enum
Class Name:         <NO CLASS>
Last Write Time:    10/30/2000 - 7:20 AM
Value 0
  Name:             0
  Type:             REG_SZ
  Data:             Root\LEGACY_GAMDRV\0000

Value 1
  Name:             Count
  Type:             REG_DWORD
  Data:             0x1

Value 2
  Name:             NextInstance
  Type:             REG_DWORD
  Data:             0x1


Key Name:           SYSTEM\CurrentControlSet\Services\gamdrv\Security
Class Name:         <NO CLASS>
Last Write Time:    10/3/2000 - 4:30 PM
Value 0
  Name:             Security
  Type:             REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 20 02 00 00  ............
...
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 00 00 00 00 00 00 18 00   ...
............
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 00 00 00 00  ....
...#........
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\gamscm

```
Key Name:           SYSTEM\CurrentControlSet\Services\gamscm
Class Name:         <NO CLASS>
Last Write Time:    10/30/2000 - 6:55 AM
Value 0
  Name:             DisplayName
  Type:             REG_SZ
  Data:             Mylex Global Array Manager Server

Value 1
  Name:             ErrorControl
  Type:             REG_DWORD
  Data:             0x1
```

```
Value 2
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            %SystemRoot%\system32\GAMSERV\gamscm.exe

Value 3
  Name:            ObjectName
  Type:            REG_SZ
  Data:            LocalSystem

Value 4
  Name:            Start
  Type:            REG_DWORD
  Data:            0x3

Value 5
  Name:            Type
  Type:            REG_DWORD
  Data:            0x10


Key Name:          SYSTEM\CurrentControlSet\Services\gamscm\Enum
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:20 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:            Root\LEGACY_GAMSCM\0000

Value 1
  Name:            Count
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\gamscm\Security
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 4:30 PM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 00 00 00 00
................
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 00 00 00 00 00 00 18 00   ...
...........
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 00 00 00 00  ....
...#.......
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\GniConMgr

```
Key Name:          SYSTEM\CurrentControlSet\Services\GniConMgr
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            DependOnGroup
  Type:            REG_MULTI_SZ
  Data:

Value 1
  Name:            DependOnService
  Type:            REG_MULTI_SZ
  Data:            GniVIA


Value 2
  Name:            DisplayName
  Type:            REG_SZ
  Data:            cLAN Connection Manager

Value 3
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 4
  Name:            Group
  Type:            REG_SZ
  Data:            TDI

Value 5
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            %SystemRoot%\System32\GnConMgr.exe

Value 6
  Name:            ObjectName
  Type:            REG_SZ
  Data:            LocalSystem

Value 7
```

```
  Name:            Start
  Type:            REG_DWORD
  Data:            0x2

Value 8
  Name:            Type
  Type:            REG_DWORD
  Data:            0x10


Key Name:          SYSTEM\CurrentControlSet\Services\GniConMgr\Enum
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:20 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:            Root\LEGACY_GNICONMGR\0000

Value 1
  Name:            Count
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\GniConMgr\Parameters
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            NicName
  Type:            REG_SZ
  Data:            GniVIA


Key Name:          SYSTEM\CurrentControlSet\Services\GniConMgr\Security
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 76 00 69 00
............v.i.
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 63 00 65 00 00 00 18 00   ...
...c.e.....
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 63 00 65 00  ....
...#...c.e.
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\GNINDIS

```
Key Name:          SYSTEM\CurrentControlSet\Services\GNINDIS
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            DisplayName
  Type:            REG_SZ
  Data:            cLAN NDIS Driver

Value 1
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            Group
  Type:            REG_SZ
  Data:            NDIS

Value 3
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            System32\DRIVERS\gnindis.sys

Value 4
  Name:            Start
  Type:            REG_DWORD
  Data:            0x2

Value 5
  Name:            Tag
  Type:            REG_DWORD
  Data:            0xc

Value 6
  Name:            Type
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\GNINDIS\Enum
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:20 AM
Value 0
  Name:            0
```

```
  Type:            REG_SZ
  Data:
PCI\VEN_135B&DEV_0001&SUBSYS_00000000&REV_00\3&13c0b0c5&0&20

Value 1
  Name:            Count
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\GNINDIS\Security
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 76 00 63 00
............v.c.
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 00 00 00 00 00 00 18 00    ...
............
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 00 00 00 00  ....
...#........
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\GNINVIPL

```
Key Name:          SYSTEM\CurrentControlSet\Services\GNINVIPL
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            DisplayName
  Type:            REG_SZ
  Data:            cLAN VIPL Driver

Value 1
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            System32\DRIVERS\gninvipl.sys

Value 3
  Name:            Start
  Type:            REG_DWORD
  Data:            0x3

Value 4
  Name:            Type
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\GNINVIPL\Security
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 6e 00 61 00
............n.a.
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 67 00 65 00 00 00 18 00    ...
...g.e.....
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 67 00 65 00  ....
...#...g.e.
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\GniVIA

```
Key Name:          SYSTEM\CurrentControlSet\Services\GniVIA
Class Name:        <NO CLASS>
```

```
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            DisplayName
  Type:            REG_SZ
  Data:            cLAN VIA Driver

Value 1
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            Group
  Type:            REG_SZ
  Data:            PNP_TDI

Value 3
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            System32\DRIVERS\GniVIA.sys

Value 4
  Name:            Start
  Type:            REG_DWORD
  Data:            0x2

Value 5
  Name:            Tag
  Type:            REG_DWORD
  Data:            0x7

Value 6
  Name:            Type
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\GniVIA\Enum
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:20 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:            Root\LEGACY_GNIVIA\0000

Value 1
  Name:            Count
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\GniVIA\Linkage
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            Bind
  Type:            REG_MULTI_SZ
  Data:            \Device\{03171A57-49B7-4F23-B82F-843F7FA01870}


Value 1
  Name:            Export
  Type:            REG_MULTI_SZ
  Data:            \Device\GniVIA_{03171A57-49B7-4F23-B82F-843F7FA01870}


Value 2
  Name:            Route
  Type:            REG_MULTI_SZ
  Data:            "{03171A57-49B7-4F23-B82F-843F7FA01870}"



Key Name:          SYSTEM\CurrentControlSet\Services\GniVIA\Parameters
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            <NO NAME>
  Type:            REG_SZ
  Data:

Value 1
  Name:            MaxNumberNics
  Type:            REG_DWORD
  Data:            0x2


Key Name:          SYSTEM\CurrentControlSet\Services\GniVIA\Performance
Class Name:        <NO CLASS>
Last Write Time:   10/30/2000 - 7:22 AM
Value 0
  Name:            Close
  Type:            REG_SZ
  Data:            CloseNicPerformanceData

Value 1
  Name:            Collect
  Type:            REG_SZ
  Data:            CollectNicPerformanceData

Value 2
  Name:            First Counter
  Type:            REG_DWORD
  Data:            0x8a4

Value 3
  Name:            First Help
  Type:            REG_DWORD
  Data:            0x8a5

Value 4
  Name:            Last Counter
  Type:            REG_DWORD
```

```
  Data:            0x8e2

Value 5
  Name:            Last Help
  Type:            REG_DWORD
  Data:            0x8e3

Value 6
  Name:            Library
  Type:            REG_SZ
  Data:            gni_perf.dll

Value 7
  Name:            Open
  Type:            REG_SZ
  Data:            OpenNicPerformanceData

Value 8
  Name:            WbemAdapFileSize
  Type:            REG_DWORD
  Data:            0xc000

Value 9
  Name:            WbemAdapFileTime
  Type:            REG_BINARY
  Data:
00000000   60 f2 55 07 f8 82 bf 01 -                           `òU.ø.¿.

Value 10
  Name:            WbemAdapStatus
  Type:            REG_DWORD
  Data:            0


Key Name:          SYSTEM\CurrentControlSet\Services\GniVIA\Security
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 8:50 AM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 00 00 00 00
................
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - c0 3b fd 64 00 00 18 00    ...
...À;ýd....
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 c0 3b fd 64  ....
...#...À;ýd
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\macdisk

```
Key Name:          SYSTEM\CurrentControlSet\Services\macdisk
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 7:47 AM
Value 0
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 1
  Name:            Group
  Type:            REG_SZ
  Data:            System Bus Extender

Value 2
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            System32\DRIVERS\mac2w2k.sys

Value 3
  Name:            Start
  Type:            REG_DWORD
  Data:            0

Value 4
  Name:            Tag
  Type:            REG_DWORD
  Data:            0x6

Value 5
  Name:            Type
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\macdisk\Enum
Class Name:        <NO CLASS>
Last Write Time:   11/1/2000 - 9:43 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:            Root\LEGACY_MACDISK\0000

Value 1
  Name:            1
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&1c8e952d&0&400

Value 2
```

```
  Name:            2
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&1c8e952d&0&410

Value 3
  Name:            3
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&1c8e952d&0&420

Value 4
  Name:            4
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&2da1b927&0&400

Value 5
  Name:            5
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&2e1f7b3c&0&400

Value 6
  Name:            6
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&2dd424b2&0&400

Value 7
  Name:            7
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&32a6693b&0&400

Value 8
  Name:            8
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&aa6c454&0&400

Value 9
  Name:            9
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&27c8c521&0&400

Value 10
  Name:            Count
  Type:            REG_DWORD
  Data:            0xa

Value 11
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0xa


Key Name:          SYSTEM\CurrentControlSet\Services\macdisk\Parameters
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 7:47 AM

Key Name:
SYSTEM\CurrentControlSet\Services\macdisk\Parameters\PnpInterface
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 7:47 AM
Value 0
  Name:            5
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\macdisk\Security
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 7:47 AM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 6f 00 72 00
............o.r.
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 74 00 00 00 00 00 18 00   ...
...t.......
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 74 00 00 00  ....
...#...t...
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\MSDTC

```
Key Name:          SYSTEM\CurrentControlSet\Services\MSDTC
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 2:00 PM
Value 0
  Name:            DependOnGroup
  Type:            REG_MULTI_SZ
  Data:

Value 1
  Name:            DependOnService
```

```
  Type:            REG_MULTI_SZ
  Data:            RPCSS
                   SamSS


Value 2
  Name:            Description
  Type:            REG_SZ
  Data:            Coordinates transactions that are distributed across
two or more databases, message queues, file systems, or other transaction
protected resource managers.

Value 3
  Name:            DisplayName
  Type:            REG_SZ
  Data:            Distributed Transaction Coordinator

Value 4
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 5
  Name:            Group
  Type:            REG_SZ
  Data:            MS Transactions

Value 6
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            C:\WINNT\System32\msdtc.exe

Value 7
  Name:            ObjectName
  Type:            REG_SZ
  Data:            LocalSystem

Value 8
  Name:            Start
  Type:            REG_DWORD
  Data:            0x3

Value 9
  Name:            Type
  Type:            REG_DWORD
  Data:            0x110


Key Name:          SYSTEM\CurrentControlSet\Services\MSDTC\Enum
Class Name:        <NO CLASS>
Last Write Time:   11/1/2000 - 9:42 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:            Root\LEGACY_MSDTC\0000

Value 1
  Name:            Count
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\MSDTC\Performance
Class Name:        <NO CLASS>
Last Write Time:   11/1/2000 - 9:44 AM
Value 0
  Name:            Close
  Type:            REG_SZ
  Data:            DtcPerfClose````Close`ƒì°

Value 1
  Name:            Collect
  Type:            REG_SZ
  Data:            DtcPerfCollect``Collect`DtcPer

Value 2
  Name:            First Counter
  Type:            REG_DWORD
  Data:            0x7e6

Value 3
  Name:            First Help
  Type:            REG_DWORD
  Data:            0x7e7

Value 4
  Name:            Last Counter
  Type:            REG_DWORD
  Data:            0x800

Value 5
  Name:            Last Help
  Type:            REG_DWORD
  Data:            0x801

Value 6
  Name:            Library
  Type:            REG_SZ
  Data:            msdtcui.DLL`Library`DtcP

Value 7
  Name:            Object List
  Type:            REG_SZ
  Data:            2022

Value 8
  Name:            Open
  Type:            REG_SZ
  Data:            DtcPerfOpen`Open````DtcP

Value 9
  Name:            WbemAdapFileSize
  Type:            REG_DWORD
```

```
  Data:            0x23510

Value 10
  Name:            WbemAdapFileTime
  Type:            REG_BINARY
  Data:
00000000   00 20 83 2d 30 01 c0 01 -                          . .-0.À.

Value 11
  Name:            WbemAdapStatus
  Type:            REG_DWORD
  Data:            0


Key Name:          SYSTEM\CurrentControlSet\Services\MSDTC\Security
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 2:40 PM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 bc 00 00 00 - c8 00 00 00 14 00 00 00
....¼...È.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 8c 00 06 00 00 00 - 00 00 14 00 8d 01 02 00
................
00000040   01 01 00 00 00 00 00 05 - 0b 00 00 00 00 00 18 00
................
00000050   9d 01 02 00 01 02 00 00 - 00 00 00 05 20 00 00 00  ............
...
00000060   23 02 00 00 00 00 18 00 - ff 01 0f 00 01 02 00 00
#.......ÿ.......
00000070   00 00 00 05 20 00 00 00 - 20 02 00 00 00 00 18 00  .... ...
........
00000080   ff 01 0f 00 01 02 00 00 - 00 00 00 05 20 00 00 00  ÿ...........
...
00000090   25 02 00 00 00 00 14 00 - fd 01 02 00 01 01 00 00
%.......ý.......
000000a0   00 00 00 05 12 00 00 00 - 00 00 14 00 10 00 00 00
................
000000b0   01 01 00 00 00 00 00 01 - 00 00 00 00 01 01 00 00
................
000000c0   00 00 00 05 12 00 00 00 - 01 01 00 00 00 00 00 05
................
000000d0   12 00 00 00                                        ....
```

CurrentControlSet\Services\MSSQLSERVER

```
Key Name:          SYSTEM\CurrentControlSet\Services\MSSQLSERVER
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            DisplayName
  Type:            REG_SZ
  Data:            MSSQLSERVER

Value 1
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            C:\PROGRA~1\MICROS~2\MSSQL\binn\sqlservr.exe

Value 3
  Name:            ObjectName
  Type:            REG_SZ
  Data:            LocalSystem

Value 4
  Name:            Start
  Type:            REG_DWORD
  Data:            0x3

Value 5
  Name:            Type
  Type:            REG_DWORD
  Data:            0x10


Key Name:          SYSTEM\CurrentControlSet\Services\MSSQLSERVER\Enum
Class Name:        <NO CLASS>
Last Write Time:   11/1/2000 - 9:42 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:            Root\LEGACY_MSSQLSERVER\0000

Value 1
  Name:            Count
  Type:            REG_DWORD
  Data:            0x1

Value 2
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\MSSQLSERVER\Linkage
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            Export
  Type:            REG_MULTI_SZ
  Data:            MSSQLSERVER



Key Name:
SYSTEM\CurrentControlSet\Services\MSSQLSERVER\Performance
```

```
Class Name:        <NO CLASS>
Last Write Time:   11/1/2000 - 9:44 AM
Value 0
  Name:            Close
  Type:            REG_SZ
  Data:            CloseSQLPerformanceData

Value 1
  Name:            Collect
  Type:            REG_SZ
  Data:            CollectSQLPerformanceData

Value 2
  Name:            First Counter
  Type:            REG_DWORD
  Data:            0x8e4

Value 3
  Name:            First Help
  Type:            REG_DWORD
  Data:            0x8e5

Value 4
  Name:            Last Counter
  Type:            REG_DWORD
  Data:            0x9f8

Value 5
  Name:            Last Help
  Type:            REG_DWORD
  Data:            0x9f9

Value 6
  Name:            Library
  Type:            REG_SZ
  Data:            C:\PROGRA~1\MICROS~2\MSSQL\BINN\SQLCTR80.DLL

Value 7
  Name:            Open
  Type:            REG_SZ
  Data:            OpenSQLPerformanceData

Value 8
  Name:            WbemAdapFileSize
  Type:            REG_DWORD
  Data:            0x803b

Value 9
  Name:            WbemAdapFileTime
  Type:            REG_BINARY
  Data:
00000000   00 b1 b4 91 83 ff bf 01 -                         .±´..ÿ¿.

Value 10
  Name:            WbemAdapStatus
  Type:            REG_DWORD
  Data:            0
```

```
Key Name:          SYSTEM\CurrentControlSet\Services\MSSQLSERVER\Security
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 00 00 00 00
................
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 00 00 00 00 00 00 18 00   ...
............
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý.......
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 00 00 00 00  ....
...#........
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

CurrentControlSet\Services\PartMgr

```
Key Name:          SYSTEM\CurrentControlSet\Services\PartMgr
Class Name:        <NO CLASS>
Last Write Time:   10/3/2000 - 7:20 AM
Value 0
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 1
  Name:            Group
  Type:            REG_SZ
  Data:            System Bus Extender

Value 2
  Name:            Start
  Type:            REG_DWORD
  Data:            0

Value 3
  Name:            Tag
  Type:            REG_DWORD
  Data:            0x5
```

```
Value 4
  Name:            Type
  Type:            REG_DWORD
  Data:            0x1


Key Name:          SYSTEM\CurrentControlSet\Services\PartMgr\Enum
Class Name:        <NO CLASS>
Last Write Time:   11/1/2000 - 9:43 AM
Value 0
  Name:            0
  Type:            REG_SZ
  Data:            Root\LEGACY_PARTMGR\0000

Value 1
  Name:            1
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_UNISYS&Prod_018200MAG3182LC&Rev_0610\4&776944c&0&000

Value 2
  Name:            10
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&27c8c521&0&400

Value 3
  Name:            2
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&1c8e952d&0&400

Value 4
  Name:            3
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&1c8e952d&0&410

Value 5
  Name:            4
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&1c8e952d&0&420

Value 6
  Name:            5
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&2da1b927&0&400

Value 7
  Name:            6
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&2e1f7b3c&0&400

Value 8
  Name:            7
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&2dd424b2&0&400

Value 9
  Name:            8
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&32a6693b&0&400

Value 10
  Name:            9
  Type:            REG_SZ
  Data:
SCSI\Disk&Ven_MYLEX&Prod_eXtremeRAID_2000&Rev_0600\5&aa6c454&0&400

Value 11
  Name:            Count
  Type:            REG_DWORD
  Data:            0xb

Value 12
  Name:            NextInstance
  Type:            REG_DWORD
  Data:            0xb
```

CurrentControlSet\Services\SQLSERVERAGENT

```
Key Name:          SYSTEM\CurrentControlSet\Services\SQLSERVERAGENT
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            DependOnGroup
  Type:            REG_MULTI_SZ
  Data:

Value 1
  Name:            DependOnService
  Type:            REG_MULTI_SZ
  Data:            MSSQLSERVER


Value 2
  Name:            DisplayName
  Type:            REG_SZ
  Data:            SQLSERVERAGENT

Value 3
  Name:            ErrorControl
  Type:            REG_DWORD
  Data:            0x1

Value 4
  Name:            ImagePath
  Type:            REG_EXPAND_SZ
  Data:            C:\PROGRA~1\MICROS~2\MSSQL\binn\sqlagent.exe
```

```
Value 5
  Name:            ObjectName
  Type:            REG_SZ
  Data:            LocalSystem

Value 6
  Name:            Start
  Type:            REG_DWORD
  Data:            0x3

Value 7
  Name:            Type
  Type:            REG_DWORD
  Data:            0x10


Key Name:
SYSTEM\CurrentControlSet\Services\SQLSERVERAGENT\Security
Class Name:        <NO CLASS>
Last Write Time:   10/4/2000 - 1:38 PM
Value 0
  Name:            Security
  Type:            REG_BINARY
  Data:
00000000   01 00 14 80 a0 00 00 00 - ac 00 00 00 14 00 00 00
.... ...¬.......
00000010   30 00 00 00 02 00 1c 00 - 01 00 00 00 02 80 14 00
0...............
00000020   ff 01 0f 00 01 01 00 00 - 00 00 00 01 00 00 00 00
ÿ...............
00000030   02 00 70 00 04 00 00 00 - 00 00 18 00 fd 01 02 00
..p.........ý...
00000040   01 01 00 00 00 00 00 05 - 12 00 00 00 63 00 6f 00
............c.o.
00000050   00 00 1c 00 ff 01 0f 00 - 01 02 00 00 00 00 00 05
....ÿ...........
00000060   20 00 00 00 20 02 00 00 - 6d 00 00 00 00 00 18 00   ...
...m........
00000070   8d 01 02 00 01 01 00 00 - 00 00 00 05 0b 00 00 00
................
00000080   20 02 00 00 00 00 1c 00 - fd 01 02 00 01 02 00 00
.......ý........
00000090   00 00 00 05 20 00 00 00 - 23 02 00 00 6d 00 00 00  ....
...#...m...
000000a0   01 01 00 00 00 00 00 05 - 12 00 00 00 01 01 00 00
................
000000b0   00 00 00 05 12 00 00 00 -                          ........
```

Windows 2000 Server Client Configuration Information

```
System Information report written at: 10/30/2000 03:26:20 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 3 GenuineIntel ~864 Mhz
Processor      x86 Family 6 Model 8 Stepping 3 GenuineIntel ~864 Mhz
BIOS Version   IBM BIOS Ver 5.0
Windows Directory      C:\WINNT
System Directory       C:\WINNT\System32
Boot Device    \Device\Harddisk0\Partition1
Locale  United States
User Name      CLIENT1\Administrator
Time Zone      Pacific Standard Time
Total Physical Memory  1,048,084 KB
Available Physical Memory     897,896 KB
Total Virtual Memory   3,570,900 KB
Available Virtual Memory      3,371,652 KB
Page File Space        2,522,816 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
0x0D00-0x4AEB  PCI bus OK
0x82E8-0xFFFF  PCI bus OK
0x03B0-0x03BB  S3 Inc. Savage4        OK
0x03C0-0x03DF  S3 Inc. Savage4        OK
0x3000-0x3FFF  DEC 21152 PCI to PCI bridge   OK
0x3100-0x311F  Intel(R) PRO/100+ Dual Port Server Adapter   OK
0x3120-0x313F  Intel(R) PRO/100+ Dual Port Server Adapter #2        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
0xFD00-0xFD3F  Motherboard resources  OK
0xFE80-0xFEBF  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
0x4AEC-0x82E7  PCI bus OK
0x4B00-0x4BFF  Adaptec AIC-7899 Ultra160/m PCI SCSI Card    OK
0x4C00-0x4CFF  Adaptec AIC-7899 Ultra160/m PCI SCSI Card    OK

[IRQs]

IRQ Number     Device
30      Microsoft ACPI-Compliant System
31      S3 Inc. Savage4
18      Intel(R) PRO/100+ Dual Port Server Adapter
19      Intel(R) PRO/100+ Dual Port Server Adapter #2
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
0xC8000-0xCFFFF        PCI bus OK
0xC8000-0xCFFFF        System board   OK
0xF0000000-0xFEBFFFFF  PCI bus OK
0xF0000000-0xFEBFFFFF  S3 Inc. Savage4        OK
0xFED00000-0xFEDFFFFF  PCI bus OK
0xFEF00000-0xFFDFFFFF  PCI bus OK
0xFEB80000-0xFEBFFFFF  S3 Inc. Savage4        OK
0xA0000-0xBFFFF        S3 Inc. Savage4        OK
0xFD000000-0xFEAFFFFF  DEC 21152 PCI to PCI bridge   OK
0xFC800000-0xFCFFFFFF  DEC 21152 PCI to PCI bridge   OK
0xFEA00000-0xFEA00FFF  Intel(R) PRO/100+ Dual Port Server Adapter   OK
0xFE800000-0xFE8FFFFF  Intel(R) PRO/100+ Dual Port Server Adapter   OK
0xFEA01000-0xFEA01FFF  Intel(R) PRO/100+ Dual Port Server Adapter #2
       OK
0xFE900000-0xFE9FFFFF  Intel(R) PRO/100+ Dual Port Server Adapter #2
       OK
0xFEC00000-0xFECFFFFF  Advanced programmable interrupt controller   OK
0xFEE00000-0xFEEFFFFF  Advanced programmable interrupt controller   OK
0xFF700000-0xFF700FFF  Standard OpenHCD USB Host Controller  OK
0xD0000-0xDFFFF        PCI bus OK
0x40000000-0xEFFFFFFF  PCI bus OK
0xEFFFF000-0xEFFFFFFF  Adaptec AIC-7899 Ultra160/m PCI SCSI Card    OK
0xEFFFE000-0xEFFFEFFF  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
0x0000-0x9FFFF System board   OK
0x100000-0xFFFFFF      System board   OK
0x1000000-0x3FFFFFFF   System board   OK
0xE0000-0xFFFFF        System board   OK
0xFFFE0000-0xFFFFFFFF  System board   OK
0xCC000-0xCFFFF        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\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\msg723.acm  Microsoft Corporation         OK
       C:\WINNT\System32\MSG723.ACM  4.4.3385       106.77 KB (109,328
bytes)  9/25/2000 9:29:57 AM
c:\winnt\system32\lhacm.acm   Microsoft Corporation         OK
       C:\WINNT\System32\LHACM.ACM   4.4.3385       33.27 KB (34,064
bytes)  9/25/2000 9:29:58 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\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\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\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

[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)  9/25/2000 9:29:57 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\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
c:\winnt\system32\msh263.drv  Microsoft Corporation         OK
       C:\WINNT\System32\MSH263.DRV  4.4.3385       252.27 KB (258,320
bytes)  9/25/2000 9:29:29 AM
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\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

[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    Not Available
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      10/29/2000 11:00:52 PM
Index   0
Service Name    PCNet5
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     Not Available
Service Name    PCnet
Driver  c:\winnt\system32\drivers\pcntn5m.sys (29968, 4.09.00)

Name    [00000001] Intel(R) PRO/1000 Gigabit Server Adapter
Adapter Type    Not Available
Product Name    Intel(R) PRO/1000 Gigabit Server Adapter
Installed       True
PNP Device ID
        PCI\VEN_8086&DEV_1000&SUBSYS_10008086&REV_03\3&267A616A&0&48
Last Reset      10/29/2000 11:00:52 PM
Index   1
Service Name    E1000
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     Not Available
Service Name    E1000
Driver  c:\winnt\system32\drivers\e1000nt5.sys (35600, 1.31.00.0096)

Name    [00000002] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type    Ethernet 802.3
Product Name    Intel(R) PRO/100+ Dual Port Server Adapter
Installed       True
PNP Device ID
        PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&AA5CF65&0&2050
Last Reset      10/29/2000 11:00:52 PM
Index   2
Service Name    E100B
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:D0:B7:81:D3:09
Service Name    E100B
IRQ Number      18
I/O Port        0x3100-0x311F
Driver  c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

Name    [00000003] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type    Ethernet 802.3
Product Name    Intel(R) PRO/100+ Dual Port Server Adapter
Installed       True
PNP Device ID
        PCI\VEN_8086&DEV_1229&SUBSYS_10F08086&REV_05\4&AA5CF65&0&2850
Last Reset      10/29/2000 11:00:52 PM
Index   3
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:B7:81:D3:0A
Service Name    E100B
IRQ Number      19
I/O Port        0x3120-0x313F
Driver  c:\winnt\system32\drivers\e100bnt5.sys (80144, 4.01.67.0000)

Name    [00000004] RAS Async Adapter
Adapter Type    Not Available
Product Name    RAS Async Adapter
Installed       True
PNP Device ID   Not Available
Last Reset      10/29/2000 11:00:52 PM
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      10/29/2000 11:00:52 PM
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     10/29/2000 11:00:52 PM
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_PTIMINIPORT\0000
Last Reset     10/29/2000 11:00:52 PM
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     10/29/2000 11:00:52 PM
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     10/29/2000 11:00:52 PM
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     10/29/2000 11:00:52 PM
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     10/29/2000 11:00:52 PM
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      10/29/2000 11:00:52 PM
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      10/29/2000 11:00:52 PM
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      10/29/2000 11:00:52 PM
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:00:08:FA
Service Name    GNINDIS
IRQ Number      20
```

```
Driver  c:\winnt\system32\drivers\gnindis.sys (21616, 4.01.00)


[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_{3DD6CAC5-33C3-4306-8A9B-85A1E470089B}]
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_{3DD6CAC5-33C3-4306-8A9B-85A1E470089B}]
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_{B81238B0-157D-439F-92E4-8774357E7EC7}]
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_{B81238B0-157D-439F-92E4-8774357E7EC7}]
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_{DD000891-41F6-4BFB-8ADA-98716E8D1B99}]
SEQPACKET 7
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_{DD000891-41F6-4BFB-8ADA-98716E8D1B99}]
DATAGRAM 7
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_{1E1D8D9C-5345-475F-86A6-7BC38C1A4C7F}]
SEQPACKET 8
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_{1E1D8D9C-5345-475F-86A6-7BC38C1A4C7F}]
DATAGRAM 8
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_{86998D6A-D20B-4FDC-8A3D-DD7357DC88AD}]
SEQPACKET 9
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_{86998D6A-D20B-4FDC-8A3D-DD7357DC88AD}]
DATAGRAM 9
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{FB7F59F3-6D4E-4241-A11E-
CBBCE24E9262}] 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{FB7F59F3-6D4E-4241-A11E-
CBBCE24E9262}] 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{CB009A9D-2A22-4BAC-BB36-
BCD3931D5EEE}] 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{CB009A9D-2A22-4BAC-BB36-
BCD3931D5EEE}] 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{3DC5073C-1ABB-4792-B907-
49409FE48B9B}] 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{3DC5073C-1ABB-4792-B907-
49409FE48B9B}] 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{865A2636-9E54-47AE-BE43-
F24E5395A3C8}] SEQPACKET 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{865A2636-9E54-47AE-BE43-
F24E5395A3C8}] 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{13E09C6C-992D-4396-AE82-
3A0615735F49}] SEQPACKET 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{13E09C6C-992D-4396-AE82-
3A0615735F49}] 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_{3DD6CAC5-33C3-4306-8A9B-
85A1E470089B}] SEQPACKET 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_{3DD6CAC5-33C3-4306-8A9B-
85A1E470089B}] 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_{B81238B0-157D-439F-92E4-
8774357E7EC7}] 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_{B81238B0-157D-439F-92E4-
8774357E7EC7}] 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_{DD000891-41F6-4BFB-8ADA-
98716E8D1B99}] SEQPACKET 1
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_{DD000891-41F6-4BFB-8ADA-
98716E8D1B99}] DATAGRAM 1
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_{1E1D8D9C-5345-475F-86A6-
7BC38C1A4C7F}] SEQPACKET 2
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_{1E1D8D9C-5345-475F-86A6-
7BC38C1A4C7F}] DATAGRAM 2
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_{86998D6A-D20B-4FDC-8A3D-
DD7357DC88AD}] SEQPACKET 3
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_{86998D6A-D20B-4FDC-8A3D-
DD7357DC88AD}] DATAGRAM 3
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_{0CDDB6E2-3D71-4337-852A-
D16E4786FAE5}] 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_{0CDDB6E2-3D71-4337-852A-
D16E4786FAE5}] 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_{9D34027C-1EF0-4304-AEC7-
496E8F334D22}] 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_{9D34027C-1EF0-4304-AEC7-
496E8F334D22}] 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     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     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\2


[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 (18,186,059,776 bytes)
Free Space     9.63 GB (10,340,302,848 bytes)
Volume Name
Volume Serial Number   E8ED5EFA
Partition      Disk #0, Partition #0
Partition Size 16.94 GB (18,186,061,824 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 SCSIBus  0
Drive SCSILogicalUnit  0
Drive SCSIPort 2
Drive SCSITargetId     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  adpu160m
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       0x4B00-0x4BFF
Driver  c:\winnt\system32\drivers\adpu160m.sys (64432, v3.10a)

Name    Adaptec AIC-7899 Ultra160/m PCI SCSI Card
Caption Adaptec AIC-7899 Ultra160/m PCI SCSI Card
Driver  adpu160m
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       0x4C00-0x4CFF
Driver  c:\winnt\system32\drivers\adpu160m.sys (64432, v3.10a)


[Printing]

Name    Port Name      Server Name
No printing information

[Problem Devices]

Device  PNP Device ID  Error Code
IBM Netfinity Fault Tolerance PCI Adapter
       PCI\VEN_1022&DEV_2000&SUBSYS_20001014&REV_44\3&267A616A&0&10
       22
Intel(R) PRO/1000 Gigabit Server Adapter
       PCI\VEN_8086&DEV_1000&SUBSYS_10008086&REV_03\3&267A616A&0&48
       22
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&267A616A&0&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
       Disabled       Stopped OK      Ignore  False   False
abp480n5       abp480n5       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
acpi    Microsoft ACPI Driver  c:\winnt\system32\drivers\acpi.sys    Kernel
Driver  True    Boot    Running OK      Normal  False   True
acpiec  ACPIEC  c:\winnt\system32\drivers\acpiec.sys  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
adpu160m       adpu160m       c:\winnt\system32\drivers\adpu160m.sys
       Kernel Driver  True    Boot    Running OK      Normal  False   True
afd     AFD Networking Support Environment
       c:\winnt\system32\drivers\afd.sys     Kernel Driver  True    Auto
       Running OK      Normal  False   True
aha154x Aha154x Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
aic116x aic116x Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
aic78u2 aic78u2 Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
aic78xx aic78xx Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
ami0nt  ami0nt  Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
amsint  amsint  Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
asc     asc     Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
asc3350p       asc3350p       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
asc3550 asc3550 Not Available  Kernel Driver  False   Disabled
       Stopped OK      Normal  False   False
asyncmac       RAS Asynchronous Media Driver
       c:\winnt\system32\drivers\asyncmac.sys       Kernel Driver  False
       Manual  Stopped OK      Normal  False   False
atapi   Standard IDE/ESDI Hard Disk Controller
       c:\winnt\system32\drivers\atapi.sys   Kernel Driver  True    Boot
       Running OK      Normal  False   True
atdisk  Atdisk  Not Available  Kernel Driver  False   Disabled
       Stopped OK      Ignore  False   False
atmarpc ATM ARP Client Protocol
       c:\winnt\system32\drivers\atmarpc.sys Kernel Driver  False   Manual
       Stopped OK      Normal  False   False
audstub Audio Stub Driver      c:\winnt\system32\drivers\audstub.sys Kernel
Driver  True    Manual  Running OK      Normal  False   True
beep    Beep    c:\winnt\system32\drivers\beep.sys    Kernel Driver  True
       System  Running OK      Normal  False   True
buslogic       BusLogic       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
cd20xrnt       cd20xrnt       Not Available  Kernel Driver  False
       Disabled       Stopped OK      Normal  False   False
cdaudio Cdaudio c:\winnt\system32\drivers\cdaudio.sys Kernel Driver  False
       System  Stopped OK      Ignore  False   False
cdfs    Cdfs    c:\winnt\system32\drivers\cdfs.sys    File System Driver
       True    Disabled       Running OK      Normal  False   True
cdrom   CD-ROM Driver  c:\winnt\system32\drivers\cdrom.sys   Kernel Driver
       True    System  Running OK      Normal  False   True
```

```
changer Changer Not Available  Kernel Driver  False   System  Stopped OK
        Ignore  False   False
cpqarray        Cpqarray        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
cpqarry2        cpqarry2        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
cpqfcalm        cpqfcalm        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
cpqfws2e        cpqfws2e        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
dac960nt        dac960nt        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
deckzpsx        deckzpsx        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
dfsdriver       DfsDriver       c:\winnt\system32\drivers\dfs.sys     File
System Driver  True    Boot    Running OK      Normal  False   True
disk    Disk Driver     c:\winnt\system32\drivers\disk.sys    Kernel Driver
        True    Boot    Running OK      Normal  False   True
diskperf        Diskperf        c:\winnt\system32\drivers\diskperf.sys
        Kernel Driver  True    Boot    Running OK      Normal  False   True
dmboot  dmboot  c:\winnt\system32\drivers\dmboot.sys  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
dmio    Logical Disk Manager Driver     c:\winnt\system32\drivers\dmio.sys
        Kernel Driver  True    Boot    Running OK      Normal  False   True
dmload  dmload  c:\winnt\system32\drivers\dmload.sys  Kernel Driver  True
        Boot    Running OK      Normal  False   True
e1000   Intel(R) PRO/1000 Gigabit Server Adapter Driver
        c:\winnt\system32\drivers\e1000nt5.sys       Kernel Driver  False
        Manual  Stopped OK      Normal  False   False
e100b   Intel PRO Adapter Driver
        c:\winnt\system32\drivers\e100bnt5.sys       Kernel Driver  True
        Manual  Running OK      Normal  False   True
efs     EFS     c:\winnt\system32\drivers\efs.sys     File System Driver
        True    Disabled        Running OK      Normal  False   True
fastfat Fastfat c:\winnt\system32\drivers\fastfat.sys File System Driver
        True    Disabled        Running OK      Normal  False   True
fd16_700        Fd16_700        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
fdc     Floppy Disk Controller Driver c:\winnt\system32\drivers\fdc.sys
        Kernel Driver  True    Manual  Running OK      Normal  False   True
fireport        fireport        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
flashpnt        flashpnt        Not Available  Kernel Driver  False
        Disabled        Stopped OK      Normal  False   False
flpydisk        Floppy Disk Driver
        c:\winnt\system32\drivers\flpydisk.sys       Kernel Driver  True
        Manual  Running OK      Normal  False   True
ftdisk  Volume Manager Driver  c:\winnt\system32\drivers\ftdisk.sys  Kernel
Driver  True    Boot    Running OK      Normal  False   True
gnindis cLAN NDIS Driver        c:\winnt\system32\drivers\gnindis.sys Kernel
Driver  True    Auto    Running OK      Normal  False   True
gninvipl        cLAN VIPL Driver
        c:\winnt\system32\drivers\gninvipl.sys       Kernel Driver  False
        Manual  Stopped OK      Normal  False   False
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
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
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
mnmdd   mnmdd   c:\winnt\system32\drivers\mnmdd.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
mraid35x        mraid35x        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
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\netdtect.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    Disabled        Running OK      Normal  False   True
null    Null    c:\winnt\system32\drivers\null.sys    Kernel Driver  True
        System  Running OK      Normal  False   True
nwlnkflt        IPX Traffic Filter Driver
        c:\winnt\system32\drivers\nwlnkflt.sys       Kernel Driver  False
        Manual  Stopped OK      Normal  False   False
nwlnkfwd        IPX Traffic Forwarder Driver
        c:\winnt\system32\drivers\nwlnkfwd.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
        Disabled        Stopped OK      Normal  False   False
pcnet   AMD PCNET Compatable Adapter Driver
        c:\winnt\system32\drivers\pcntn5m.sys Kernel Driver  False   Manual
        Stopped OK      Normal  False   False
pdcomp  PDCOMP  Not Available   Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False
pdframe PDFRAME Not Available   Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False
pdreli  PDRELI  Not Available   Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False
pdrframe        PDRFRAME        Not Available   Kernel Driver  False   Manual
        Stopped OK      Ignore  False   False
pptpminiport    WAN Miniport (PPTP)
        c:\winnt\system32\drivers\raspptp.sys Kernel Driver  True    Manual
        Running OK      Normal  False   True
ptilink Direct Parallel Link Driver
        c:\winnt\system32\drivers\ptilink.sys Kernel Driver  True    Manual
        Running OK      Normal  False   True
ql1080  ql1080  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
ql10wnt Ql10wnt Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
ql1240  ql1240  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
ql2100  ql2100  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
rasacd  Remote Access Auto Connection Driver
        c:\winnt\system32\drivers\rasacd.sys  Kernel Driver  True    System
        Running OK      Normal  False   True
rasl2tp WAN Miniport (L2TP)    c:\winnt\system32\drivers\rasl2tp.sys Kernel
Driver  True    Manual  Running OK      Normal  False   True
raspti  Direct Parallel        c:\winnt\system32\drivers\raspti.sys  Kernel
Driver  True    Manual  Running OK      Normal  False   True
rca     Microsoft Streaming Network Raw Channel Access
        c:\winnt\system32\drivers\rca.sys     Kernel Driver  False   Manual
        Stopped OK      Normal  False   False
rdbss   Rdbss   c:\winnt\system32\drivers\rdbss.sys   File System Driver
        True    System  Running OK      Normal  False   True
rdpwd   RDPWD   c:\winnt\system32\drivers\rdpwd.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
redbook Digital CD Audio Playback Filter Driver
        c:\winnt\system32\drivers\redbook.sys Kernel Driver  False   System
        Stopped OK      Normal  False   False
s3inc   S3Inc   c:\winnt\system32\drivers\s3sav4m.sys Kernel Driver  True
        Manual  Running OK      Ignore  False   True
serenum Serenum Filter Driver  c:\winnt\system32\drivers\serenum.sys Kernel
Driver  True    Manual  Running OK      Normal  False   True
serial  Serial port driver     c:\winnt\system32\drivers\serial.sys  Kernel
Driver  True    System  Running OK      Ignore  False   True
sfloppy Sfloppy c:\winnt\system32\drivers\sfloppy.sys Kernel Driver  False
        System  Stopped OK      Ignore  False   False
sglfb   sglfb   Not Available  Kernel Driver  False   System  Stopped OK
        Normal  False   False
simbad  Simbad  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
sparrow Sparrow Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
spud    Special Purpose Utility Driver
        c:\winnt\system32\drivers\spud.sys    Kernel Driver  True    Manual
        Running OK      Normal  False   True
srv     Srv     c:\winnt\system32\drivers\srv.sys     File System Driver
        True    Manual  Running OK      Normal  False   True
swenum  Software Bus Driver    c:\winnt\system32\drivers\swenum.sys  Kernel
Driver  True    Manual  Running OK      Normal  False   True
symc810 symc810 Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
symc8xx symc8xx Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
```

```
sym_hi  sym_hi  Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
tcpip   TCP/IP Protocol Driver c:\winnt\system32\drivers\tcpip.sys   Kernel
Driver  True    System  Running OK      Normal  False   True
tdasync TDASYNC c:\winnt\system32\drivers\tdasync.sys Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdipx   TDIPX   c:\winnt\system32\drivers\tdipx.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdnetb  TDNETB  c:\winnt\system32\drivers\tdnetb.sys  Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdpipe  TDPIPE  c:\winnt\system32\drivers\tdpipe.sys  Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdspx   TDSPX   c:\winnt\system32\drivers\tdspx.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
tdtcp   TDTCP   c:\winnt\system32\drivers\tdtcp.sys   Kernel Driver  False
        Manual  Stopped OK      Ignore  False   False
termdd  Terminal Device Driver c:\winnt\system32\drivers\termdd.sys  Kernel
Driver  False   Disabled        Stopped OK      Normal  False   False
tga     tga     Not Available  Kernel Driver  False   System  Stopped OK
        Ignore  False   False
udfs    Udfs    c:\winnt\system32\drivers\udfs.sys    File System Driver
        False   Disabled        Stopped OK      Normal  False   False
ultra66 ultra66 Not Available  Kernel Driver  False   Disabled
        Stopped OK      Normal  False   False
update  Microcode Update Driver        c:\winnt\system32\drivers\update.sys
        Kernel Driver  True    Manual  Running OK      Normal  False   True
usbhub  Microsoft USB Standard Hub Driver
        c:\winnt\system32\drivers\usbhub.sys  Kernel Driver  True    Manual
        Running OK      Normal  False   True
vgasave VgaSave c:\winnt\system32\drivers\vga.sys     Kernel Driver  True
        System  Running OK      Ignore  False   True
wanarp  Remote Access IP ARP Driver    c:\winnt\system32\drivers\wanarp.sys
        Kernel Driver  True    Manual  Running OK      Normal  False   True
wdica   WDICA   Not Available  Kernel Driver  False   Manual  Stopped OK
        Ignore  False   False

[Environment Variables]

Variable        Value   User Name
ComSpec %SystemRoot%\system32\cmd.exe <SYSTEM>
Os2LibPath      %SystemRoot%\system32\os2\dll;        <SYSTEM>
Path
        %SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;C:\P
rogram Files\Microsoft SQL Server\80\Tools\BINN      <SYSTEM>
windir  %SystemRoot%    <SYSTEM>
OS      Windows_NT      <SYSTEM>
PROCESSOR_ARCHITECTURE x86      <SYSTEM>
PROCESSOR_LEVEL         6       <SYSTEM>
PROCESSOR_IDENTIFIER    x86 Family 6 Model 8 Stepping 3, GenuineIntel
        <SYSTEM>
PROCESSOR_REVISION      0803    <SYSTEM>
NUMBER_OF_PROCESSORS    2       <SYSTEM>
PATHEXT .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH     <SYSTEM>
TEMP    %SystemRoot%\TEMP       <SYSTEM>
TMP     %SystemRoot%\TEMP       <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 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     Start Time      Version Size    File Date
system idle process     Not Available   0       0       Not Available   Not
Available       Not Available   Unknown Unknown Unknown
system  Not Available   8       8       0       1413120 Not Available
        Unknown Unknown Unknown
smss.exe        c:\winnt\system32\smss.exe      180     11      204800
        1413120 10/30/2000 7:01:19 AM   5.00.2170.1     44.27 KB (45,328
bytes)  12/7/1999 4:00:00 AM
csrss.exe       Not Available   204     13      Not Available   Not Available
        10/30/2000 7:01:21 AM   Unknown Unknown Unknown
winlogon.exe    c:\winnt\system32\winlogon.exe          200     13      204800
        1413120 10/30/2000 7:01:23 AM   5.00.2182.1     173.27 KB (177,424
bytes)  12/7/1999 4:00:00 AM
services.exe    c:\winnt\system32\services.exe          252     9       204800
        1413120 10/30/2000 7:01:23 AM   5.00.2134.1     86.77 KB (88,848
bytes)  12/7/1999 4:00:00 AM
lsass.exe       c:\winnt\system32\lsass.exe    264     13      204800
        1413120 10/30/2000 7:01:23 AM   5.00.2184.1     32.77 KB (33,552
bytes)  12/7/1999 4:00:00 AM
gnconmgr.exe    c:\winnt\system32\gnconmgr.exe          380     8       204800
        1413120 10/30/2000 7:01:25 AM   4.01.00 128.05 KB (131,119 bytes)
        9/25/2000 1:23:34 PM
svchost.exe     c:\winnt\system32\svchost.exe 440      8       204800
        1413120 10/30/2000 7:01:26 AM   5.00.2134.1     7.77 KB (7,952 bytes)
        12/7/1999 4:00:00 AM
msdtc.exe       c:\winnt\system32\msdtc.exe    480     8       204800
        1413120 10/30/2000 7:01:30 AM   1999.9.3421.3   6.77 KB (6,928 bytes)
        9/25/2000 2:26:39 AM
svchost.exe     c:\winnt\system32\svchost.exe 608      8       204800
        1413120 10/30/2000 7:01:34 AM   5.00.2134.1     7.77 KB (7,952 bytes)
        12/7/1999 4:00:00 AM
regsvc.exe      c:\winnt\system32\regsvc.exe  632      8       204800
        1413120 10/30/2000 7:01:34 AM   5.00.2155.1     65.27 KB (66,832
bytes)  12/7/1999 4:00:00 AM
mstask.exe      c:\winnt\system32\mstask.exe  652      8       204800
        1413120 10/30/2000 7:01:35 AM   4.71.2137.1     115.27 KB (118,032
bytes)  9/25/2000 9:29:47 AM
```

```
winmgmt.exe    c:\winnt\system32\wbem\winmgmt.exe    708    8      204800
       1413120 10/30/2000 7:01:35 AM  1.50.1085.0001 188.05 KB (192,567
bytes)  12/7/1999 4:00:00 AM
svchost.exe    c:\winnt\system32\svchost.exe 920    8      204800
       1413120 10/30/2000 7:01:50 AM  5.00.2134.1    7.77 KB (7,952 bytes)
       12/7/1999 4:00:00 AM
explorer.exe   c:\winnt\explorer.exe  940    8      204800  1413120
       10/30/2000 7:15:07 AM  5.00.2920.0000 232.77 KB (238,352 bytes)
       12/7/1999 4:00:00 AM
rhostsvr.exe   c:\rhost\rhostsvr.exe  788    8      204800  1413120
       10/30/2000 7:15:09 AM  Not Available  54.50 KB (55,808 bytes)
       9/25/2000 11:26:52 AM
dllhost.exe    Not Available  960    8      Not Available  Not Available
       10/30/2000 7:25:27 AM  Unknown Unknown Unknown
msinfo32.exe   c:\program files\common files\microsoft
shared\msinfo\msinfo32.exe    1216    8      204800  1413120 10/30/2000
3:25:45 PM     5.00.2134.1    15.77 KB (16,144 bytes)       9/25/2000
9:29:54 AM
rsvp.exe       c:\winnt\system32\rsvp.exe    1184    8      204800
       1413120 10/30/2000 3:26:05 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
rsvp.exe       5.00.2167.1    172.77 KB (176,912 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rsvp.exe
wbemprox.dll   1.50.1085.0001 40.05 KB (41,016 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\wbem\wbemprox.dll
cabinet.dll    5.00.2147.1    54.77 KB (56,080 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\cabinet.dll
msinfo32.dll   5.00.2177.1    312.27 KB (319,760 bytes)     9/25/2000
9:29:54 AM     Microsoft Corporation  c:\program files\common
files\microsoft shared\msinfo\msinfo32.dll
mfc42u.dll     6.00.8665.0    972.05 KB (995,384 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\mfc42u.dll
msinfo32.exe   5.00.2134.1    15.77 KB (16,144 bytes)       9/25/2000
9:29:54 AM     Microsoft Corporation  c:\program files\common
files\microsoft shared\msinfo\msinfo32.exe
rhostsvr.exe   Not Available  54.50 KB (55,808 bytes)       9/25/2000
11:26:52 AM    Not Available  c:\rhost\rhostsvr.exe
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
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
tapisrv.dll    5.00.2186.1    168.77 KB (172,816 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\tapisrv.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
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)       9/25/2000
9:29:46 AM     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)     9/25/2000
9:29:47 AM     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
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
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
rasmans.dll    5.00.2188.1    146.77 KB (150,288 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasmans.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)     9/25/2000
2:26:40 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)       9/25/2000
2:26:40 AM     Microsoft Corporation  c:\winnt\system32\xolehlp.dll
msdtclog.dll   1999.9.3421.3  89.77 KB (91,920 bytes)       9/25/2000
2:26:39 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)     9/25/2000
2:26:40 AM     Microsoft Corporation  c:\winnt\system32\msdtcprx.dll
txfaux.dll     1999.9.3422.24 341.27 KB (349,456 bytes)     9/25/2000
2:26:40 AM     Microsoft Corporation  c:\winnt\system32\txfaux.dll
msdtctm.dll    1999.9.3422.12 1.02 MB (1,070,864 bytes)     9/25/2000
2:26:40 AM     Microsoft Corporation  c:\winnt\system32\msdtctm.dll
msdtc.exe      1999.9.3421.3  6.77 KB (6,928 bytes)  9/25/2000 2:26:39 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
       Microsoft Corporation  c:\winnt\system32\wshnetbs.dll
rasadhlp.dll   5.00.2168.1    7.27 KB (7,440 bytes)  12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\rasadhlp.dll
winrnr.dll     5.00.2160.1    18.77 KB (19,216 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\winrnr.dll
dhcpcsvc.dll   5.00.2153.1    88.77 KB (90,896 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\dhcpcsvc.dll
tapi32.dll     5.00.2182.1    123.27 KB (126,224 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\tapi32.dll
rasman.dll     5.00.2188.1    54.77 KB (56,080 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasman.dll
rasapi32.dll   5.00.2188.1    189.77 KB (194,320 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rasapi32.dll
iphlpapi.dll   5.00.2173.2    67.77 KB (69,392 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\iphlpapi.dll
rnr20.dll      5.00.2152.1    35.77 KB (36,624 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rnr20.dll
wshtcpip.dll   5.00.2134.1    17.27 KB (17,680 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\wshtcpip.dll
msafd.dll      5.00.2153.1    54.27 KB (55,568 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\msafd.dll
rpcss.dll      5.00.2181.1    229.27 KB (234,768 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rpcss.dll
svchost.exe    5.00.2134.1    7.77 KB (7,952 bytes)  12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\svchost.exe
vipl.dll       4.01.00 80.00 KB (81,920 bytes)       9/25/2000 1:48:28 PM
       Giganet Incorporated   c:\winnt\system32\vipl.dll
gnconmgr.exe   4.01.00 128.05 KB (131,119 bytes)     9/25/2000 1:23:34 PM
       Giganet Incorporated   c:\winnt\system32\gnconmgr.exe
iissuba.dll    5.00.0984      9.77 KB (10,000 bytes) 12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\iissuba.dll
scecli.dll     5.00.2191.1    105.27 KB (107,792 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\scecli.dll
atl.dll 3.00.8449      57.56 KB (58,938 bytes)       12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\atl.dll
certcli.dll    5.00.2175.1    132.27 KB (135,440 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\certcli.dll
esent.dll      6.0.3939.6     1.07 MB (1,120,016 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\esent.dll
mswsock.dll    5.00.2152.1    62.27 KB (63,760 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\mswsock.dll
ntdsatq.dll    5.00.2181.1    31.27 KB (32,016 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntdsatq.dll
ntdsa.dll      5.00.2195.1    993.27 KB (1,017,104 bytes)   12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\ntdsa.dll
kdcsvc.dll     5.00.2181.1    133.77 KB (136,976 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\kdcsvc.dll
sfmapi.dll     5.00.2134.1    38.77 KB (39,696 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\sfmapi.dll
rtutils.dll    5.00.2168.1    43.77 KB (44,816 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rtutils.dll
adsldpc.dll    5.00.2172.1    127.77 KB (130,832 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\adsldpc.dll
activeds.dll   5.00.2172.1    172.77 KB (176,912 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\activeds.dll
mprapi.dll     5.00.2181.1    79.27 KB (81,168 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\mprapi.dll
rassfm.dll     5.00.2168.1    21.27 KB (21,776 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\rassfm.dll
mpr.dll 5.00.2146.1    53.27 KB (54,544 bytes)       12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\mpr.dll
schannel.dll   5.00.2170.1    139.77 KB (143,120 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\schannel.dll
netlogon.dll   5.00.2182.1    347.77 KB (356,112 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\netlogon.dll
msv1_0.dll     5.00.2164.1    94.77 KB (97,040 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\msv1_0.dll
kerberos.dll   5.00.2181.1    196.77 KB (201,488 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\kerberos.dll
msprivs.dll    5.00.2154.1    41.50 KB (42,496 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\msprivs.dll
samsrv.dll     5.00.2192.1    357.77 KB (366,352 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\samsrv.dll
lsasrv.dll     5.00.2184.1    487.77 KB (499,472 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\lsasrv.dll
lsass.exe      5.00.2184.1    32.77 KB (33,552 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\lsass.exe
ntlsapi.dll    5.00.2134.1    6.77 KB (6,928 bytes)  12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\ntlsapi.dll
wmicore.dll    5.00.2178.1    70.77 KB (72,464 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\wmicore.dll
browser.dll    5.00.2142.1    48.27 KB (49,424 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\browser.dll
seclogon.dll   5.00.2135.1    15.77 KB (16,144 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\seclogon.dll
psbase.dll     5.00.2146.1    111.77 KB (114,448 bytes)     12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\psbase.dll
cryptsvc.dll   5.00.2181.1    61.77 KB (63,248 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\cryptsvc.dll
cryptdll.dll   5.00.2135.1    41.27 KB (42,256 bytes)       12/7/1999
4:00:00 AM     Microsoft Corporation  c:\winnt\system32\cryptdll.dll
```

```
wkssvc.dll      5.00.2181.1    95.27 KB (97,552 bytes)       12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\wkssvc.dll
srvsvc.dll      5.00.2178.1    79.27 KB (81,168 bytes)       12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\srvsvc.dll
cfgmgr32.dll    5.00.2134.1    16.77 KB (17,168 bytes)       12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\cfgmgr32.dll
dmserver.dll    2191.1.296.2   11.77 KB (12,048 bytes)       12/7/1999
4:00:00 AM      VERITAS Software Corp. c:\winnt\system32\dmserver.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
icmp.dll        5.00.2134.1    7.27 KB (7,440 bytes)  12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\icmp.dll
lmhsvc.dll      5.00.2134.1    9.27 KB (9,488 bytes)  12/7/1999 4:00:00 AM
       Microsoft Corporation  c:\winnt\system32\lmhsvc.dll
eventlog.dll    5.00.2178.1    43.77 KB (44,816 bytes)       12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\eventlog.dll
ntdsapi.dll     5.00.2160.1    56.27 KB (57,616 bytes)       12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\ntdsapi.dll
scesrv.dll      5.00.2188.1    225.77 KB (231,184 bytes)     12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\scesrv.dll
umpnpmgr.dll    5.00.2182.1    86.27 KB (88,336 bytes)       12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\umpnpmgr.dll
services.exe    5.00.2134.1    86.77 KB (88,848 bytes)       12/7/1999
4:00:00 AM      Microsoft Corporation  c:\winnt\system32\services.exe
clbcatq.dll     1999.9.3422.14 479.27 KB (490,768 bytes)     9/25/2000
2:26:34 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
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
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
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
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
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
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
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
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Application Management AppMgmt Stopped Manual  Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Computer Browser       Browser Running Auto    Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Indexing Service       cisvc   Stopped Manual  Share Process
       c:\winnt\system32\cisvc.exe   Normal  LocalSystem    0
ClipBook       ClipSrv Stopped Manual  Own Process
       c:\winnt\system32\clipsrv.exe Normal  LocalSystem    0
Distributed File System       Dfs     Stopped Disabled       Own Process
       c:\winnt\system32\dfssvc.exe  Normal  LocalSystem    0
DHCP Client    Dhcp    Stopped Disabled       Share Process
       c:\winnt\system32\services.exe        Normal  LocalSystem    0
Logical Disk Manager Administrative Service  dmadmin Stopped Manual  Share
Process c:\winnt\system32\dmadmin.exe /com    Normal  LocalSystem    0
Logical Disk Manager   dmserver       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        Stopped Auto    Share Process
       c:\winnt\system32\inetsrv\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      MSIServer       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 Running Manual  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\inetsrv\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   C:\WINNT\System32\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     51876-270-8624215-05720
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 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
advpack.dll    5.0.2920.0     87 KB   12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
browselc.dll   5.0.2920.0     35 KB   12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
browseui.dll   5.0.2920.0     793 KB  12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
ckcnv.exe      5.0.2189.1     9 KB    12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
comctl32.dll   5.81.2920.0    540 KB  12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
crypt32.dll    5.131.2173.1   466 KB  12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
enhsig.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 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
iexplore.exe   5.0.2920.0     59 KB  12/7/1999 4:00:00 AM  C:\Program
Files\Internet Explorer       Microsoft Corporation
imagehlp.dll   5.0.2195.1     125 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
imghelp.dll    <File Missing> Not Available  Not Available  Not Available
       Not Available
inseng.dll     5.0.2920.0     72 KB  12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
jobexec.dll    5.0.0.1 47 KB  12/7/1999 4:00:00 AM  C:\WINNT\system32
       Microsoft Corporation
jscript.dll    5.1.0.4615     476 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
jsproxy.dll    5.0.2920.0     13 KB  12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
msaahtml.dll   <File Missing> Not Available  Not Available  Not Available
       Not Available
mshtml.dll     5.0.2920.0     2302 KB12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
msjava.dll     5.0.3234.0     918 KB 12/7/1999 4:00:00 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 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
occache.dll    5.0.2920.0     86 KB  12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
ole32.dll      5.0.2181.1     966 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
oleaut32.dll   2.40.4512.1    600 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
olepro32.dll   5.0.4512.1     160 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
rsabase.dll    5.0.2150.1     129 KB 12/7/1999 4:00:00 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 4:00:00 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 KB12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
shell32.dll    5.0.2920.0     2297 KB12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
shlwapi.dll    5.0.2920.0     283 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
url.dll5.0.2920.0     82 KB  12/7/1999 4:00:00 AM  C:\WINNT\system32
       Microsoft Corporation
urlmon.dll     5.0.2920.0     427 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
vbscript.dll   5.1.0.4615     428 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
webcheck.dll   5.0.2920.0     252 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
win.com 5.0.2134.1     24 KB  12/7/1999 4:00:00 AM  C:\WINNT\system32
       Microsoft Corporation
wininet.dll    5.0.2920.0     457 KB 12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
winsock.dll    3.10.0.103     3 KB   12/7/1999 4:00:00 AM
       C:\WINNT\system32      Microsoft Corporation
wintrust.dll   5.131.2143.1   162 KB 12/7/1999 4:00:00 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 4:00:00 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       17343 MB
Available Disk Space   9861 MB
Maximum Cache Size     541 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  9/25/2000 to 9/1/2100  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 0xbeb34542
0x01c026d2 0x00002510 0x00000000
        WbemAdapFileTime = REG_BINARY 0x00000008 0x964e6000 0x01bf40aa
        WbemAdapFileSize = REG_DWORD 0x00002510
        WbemAdapStatus = REG_DWORD 0x00000000
```

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\inetsrv\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,,201
            /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
        WbemAdapFileTime = 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\unisys
    TPCC
        MAXTERMS = 20000
        DELIVERYTHREADS = 10
        SERVERNAME = DBSERVER
        DATABASE = tpcc

\registry\machine\software\microsoft\mssqlserver\client\connectto
    DBSERVER = DBNETLIB,via:CAPRICORN8,1433,0
```

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 "webnnn.cmd <cfg file suffix> <min driver #> <max driver #>"
pause

:end
```

Profiles used for Performance Run

Web4896.cfg

```
//
//  Common Driver Configuration
//
INITBASEPORT 4300
INITSYNCMAX 4
INITPAUSE 1
INITRSCALE 350
INITTSCALE 100
INITRWID 1, 4896
INITFIXEDWID 1
INITCCLAST 208
INITCCID 208
INITCITEMID 208
//
//  Configuration Driver 1
//
1 INITIPADDR 192.168.90.31
1 INITIISADDR 192.10.1.1
1 INITIISPORT 80
1 INITBROWSERS 1020
1 INITMYWID 1,102

//  Configuration Driver 2
//
2 INITIPADDR 192.168.90.31
2 INITIISADDR 192.10.2.1
2 INITIISPORT 80
2 INITBROWSERS 1020
2 INITMYWID 103,204

//  Configuration Driver 3
//
3 INITIPADDR 192.168.90.31
3 INITIISADDR 192.10.3.1
3 INITIISPORT 80
3 INITBROWSERS 1020
3 INITMYWID 205,306

//  Configuration Driver 4
//
4 INITIPADDR 192.168.90.31
4 INITIISADDR 192.10.4.1
4 INITIISPORT 80
4 INITBROWSERS 1020
4 INITMYWID 307,408

//  Configuration Driver 5
//
5 INITIPADDR 192.168.90.31
5 INITIISADDR 192.10.5.1
5 INITIISPORT 80
5 INITBROWSERS 1020
5 INITMYWID 409,510

//  Configuration Driver 6
//
6 INITIPADDR 192.168.90.31
6 INITIISADDR 192.10.6.1
6 INITIISPORT 80
6 INITBROWSERS 1020
6 INITMYWID 511,612

//  Configuration Driver 7
//
7 INITIPADDR 192.168.90.31
7 INITIISADDR 192.10.7.1
7 INITIISPORT 80
7 INITBROWSERS 1020
7 INITMYWID 613,714

//  Configuration Driver 8
//
```

```
8 INITIPADDR 192.168.90.31
8 INITIISADDR 192.10.8.1
8 INITIISPORT 80
8 INITBROWSERS 1020
8 INITMYWID 715,816

//  Configuration Driver 9
//
9 INITIPADDR 192.168.90.32
9 INITIISADDR 192.10.9.1
9 INITIISPORT 80
9 INITBROWSERS 1020
9 INITMYWID 817,918

//  Configuration Driver 10
//
10 INITIPADDR 192.168.90.32
10 INITIISADDR 192.10.10.1
10 INITIISPORT 80
10 INITBROWSERS 1020
10 INITMYWID 919,1020

//  Configuration Driver 11
//
11 INITIPADDR 192.168.90.32
11 INITIISADDR 192.10.11.1
11 INITIISPORT 80
11 INITBROWSERS 1020
11 INITMYWID 1021,1122

//  Configuration Driver 12
//
12 INITIPADDR 192.168.90.32
12 INITIISADDR 192.10.12.1
12 INITIISPORT 80
12 INITBROWSERS 1020
12 INITMYWID 1123,1224

//  Configuration Driver 13
//
13 INITIPADDR 192.168.90.32
13 INITIISADDR 192.10.13.1
13 INITIISPORT 80
13 INITBROWSERS 1020
13 INITMYWID 1225,1326

//  Configuration Driver 14
//
14 INITIPADDR 192.168.90.32
14 INITIISADDR 192.10.14.1
14 INITIISPORT 80
14 INITBROWSERS 1020
14 INITMYWID 1327,1428

//  Configuration Driver 15
//
15 INITIPADDR 192.168.90.32
15 INITIISADDR 192.10.15.1
15 INITIISPORT 80
15 INITBROWSERS 1020
15 INITMYWID 1429,1530

//  Configuration Driver 16
//
16 INITIPADDR 192.168.90.32
16 INITIISADDR 192.10.16.1
16 INITIISPORT 80
16 INITBROWSERS 1020
16 INITMYWID 1531,1632

//  Configuration Driver 17
//
17 INITIPADDR 192.168.90.33
17 INITIISADDR 192.20.1.2
17 INITIISPORT 80
17 INITBROWSERS 1020
17 INITMYWID 1633,1734

//  Configuration Driver 18
//
18 INITIPADDR 192.168.90.33
18 INITIISADDR 192.20.2.2
18 INITIISPORT 80
18 INITBROWSERS 1020
18 INITMYWID 1735,1836

//  Configuration Driver 19
//
19 INITIPADDR 192.168.90.33
19 INITIISADDR 192.20.3.2
19 INITIISPORT 80
19 INITBROWSERS 1020
19 INITMYWID 1837,1938

//  Configuration Driver 20
//
20 INITIPADDR 192.168.90.33
20 INITIISADDR 192.20.4.2
20 INITIISPORT 80
20 INITBROWSERS 1020
20 INITMYWID 1939,2040

//  Configuration Driver 21
//
21 INITIPADDR 192.168.90.33
21 INITIISADDR 192.20.5.2
21 INITIISPORT 80
21 INITBROWSERS 1020
21 INITMYWID 2041,2142

//  Configuration Driver 22
//
22 INITIPADDR 192.168.90.33
22 INITIISADDR 192.20.6.2
22 INITIISPORT 80
22 INITBROWSERS 1020
22 INITMYWID 2143,2244
```

```
//  Configuration Driver 23
//
23 INITIPADDR 192.168.90.33
23 INITIISADDR 192.20.7.2
23 INITIISPORT 80
23 INITBROWSERS 1020
23 INITMYWID 2245,2346

//  Configuration Driver 24
//
24 INITIPADDR 192.168.90.33
24 INITIISADDR 192.20.8.2
24 INITIISPORT 80
24 INITBROWSERS 1020
24 INITMYWID 2347,2448

//  Configuration Driver 25
//
25 INITIPADDR 192.168.90.34
25 INITIISADDR 192.20.9.2
25 INITIISPORT 80
25 INITBROWSERS 1020
25 INITMYWID 2449,2550

//  Configuration Driver 26
//
26 INITIPADDR 192.168.90.34
26 INITIISADDR 192.20.10.2
26 INITIISPORT 80
26 INITBROWSERS 1020
26 INITMYWID 2551,2652

//  Configuration Driver 27
//
27 INITIPADDR 192.168.90.34
27 INITIISADDR 192.20.11.2
27 INITIISPORT 80
27 INITBROWSERS 1020
27 INITMYWID 2653,2754

//  Configuration Driver 28
//
28 INITIPADDR 192.168.90.34
28 INITIISADDR 192.20.12.2
28 INITIISPORT 80
28 INITBROWSERS 1020
28 INITMYWID 2755,2856

//  Configuration Driver 29
//
29 INITIPADDR 192.168.90.34
29 INITIISADDR 192.20.13.2
29 INITIISPORT 80
29 INITBROWSERS 1020
29 INITMYWID 2857,2958

//  Configuration Driver 30
//
30 INITIPADDR 192.168.90.34
30 INITIISADDR 192.20.14.2
30 INITIISPORT 80
30 INITBROWSERS 1020
30 INITMYWID 2959,3060

//  Configuration Driver 31
//
31 INITIPADDR 192.168.90.34
31 INITIISADDR 192.20.15.2
31 INITIISPORT 80
31 INITBROWSERS 1020
31 INITMYWID 3061,3162

//  Configuration Driver 32
//
32 INITIPADDR 192.168.90.34
32 INITIISADDR 192.20.16.2
32 INITIISPORT 80
32 INITBROWSERS 1020
32 INITMYWID 3163,3264

//  Configuration Driver 33
//
33 INITIPADDR 192.168.90.35
33 INITIISADDR 192.30.1.3
33 INITIISPORT 80
33 INITBROWSERS 1020
33 INITMYWID 3265,3366

//  Configuration Driver 34
//
34 INITIPADDR 192.168.90.35
34 INITIISADDR 192.30.2.3
34 INITIISPORT 80
34 INITBROWSERS 1020
34 INITMYWID 3367,3468

//  Configuration Driver 35
//
35 INITIPADDR 192.168.90.35
35 INITIISADDR 192.30.3.3
35 INITIISPORT 80
35 INITBROWSERS 1020
35 INITMYWID 3469,3570

//  Configuration Driver 36
//
36 INITIPADDR 192.168.90.35
36 INITIISADDR 192.30.4.3
36 INITIISPORT 80
36 INITBROWSERS 1020
36 INITMYWID 3571,3672

//  Configuration Driver 37
//
37 INITIPADDR 192.168.90.35
37 INITIISADDR 192.30.5.3
37 INITIISPORT 80
37 INITBROWSERS 1020
```

```
37 INITMYWID 3673,3774

//  Configuration Driver 38
//
38 INITIPADDR 192.168.90.35
38 INITIISADDR 192.30.6.3
38 INITIISPORT 80
38 INITBROWSERS 1020
38 INITMYWID 3775,3876

//  Configuration Driver 39
//
39 INITIPADDR 192.168.90.35
39 INITIISADDR 192.30.7.3
39 INITIISPORT 80
39 INITBROWSERS 1020
39 INITMYWID 3877,3978

//  Configuration Driver 40
//
40 INITIPADDR 192.168.90.35
40 INITIISADDR 192.30.8.3
40 INITIISPORT 80
40 INITBROWSERS 1020
40 INITMYWID 3979,4080

//  Configuration Driver 41
//
41 INITIPADDR 192.168.90.36
41 INITIISADDR 192.30.9.3
41 INITIISPORT 80
41 INITBROWSERS 1020
41 INITMYWID 4081,4182

//  Configuration Driver 42
//
42 INITIPADDR 192.168.90.36
42 INITIISADDR 192.30.10.3
42 INITIISPORT 80
42 INITBROWSERS 1020
42 INITMYWID 4183,4284

//  Configuration Driver 43
//
43 INITIPADDR 192.168.90.36
43 INITIISADDR 192.30.11.3
43 INITIISPORT 80
43 INITBROWSERS 1020
43 INITMYWID 4285,4386

//  Configuration Driver 44
//
44 INITIPADDR 192.168.90.36
44 INITIISADDR 192.30.12.3
44 INITIISPORT 80
44 INITBROWSERS 1020
44 INITMYWID 4387,4488

//  Configuration Driver 45
//
45 INITIPADDR 192.168.90.36
45 INITIISADDR 192.30.13.3
45 INITIISPORT 80
45 INITBROWSERS 1020
45 INITMYWID 4489,4590

//  Configuration Driver 46
//
46 INITIPADDR 192.168.90.36
46 INITIISADDR 192.30.14.3
46 INITIISPORT 80
46 INITBROWSERS 1020
46 INITMYWID 4591,4692

//  Configuration Driver 47
//
47 INITIPADDR 192.168.90.36
47 INITIISADDR 192.30.15.3
47 INITIISPORT 80
47 INITBROWSERS 1020
47 INITMYWID 4693,4794

//  Configuration Driver 48
//
48 INITIPADDR 192.168.90.36
48 INITIISADDR 192.30.16.3
48 INITIISPORT 80
48 INITBROWSERS 1020
48 INITMYWID 4795,4896
```

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
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
```

Appendix E - Disk Storage

TPC-C 180-Day Disk Space Requirements

Warehouses	**4896**	tpmC	**61,390.43**	tpmC/W	12.54

Table	**Initial Rows**	**Data KB**	**Index KB**	**Extra 5% KB**	**Total With 5% KB**
Warehouse	4,896	528	56	29	613
District	48,960	5,440	72	276	5,788
Customer	146,880,000	30,129,232	6,217,184	1,817,321	38,163,737
Customer.Text	146,880,000	87,853,664		4,392,683	92,246,347
History (D)	146,880,000	8,160,000	29,632		8,189,632
Order (D)	146,880,000	4,502,072	2,270,800		6,772,872
New-Order	44,064,000	696,672	1,584	34,913	733,169
Order-Line (D)	1,468,798,778	91,799,928	194,288		91,994,216
Item	100,000	9,528	72	480	10,080
Stock	489,600,000	156,672,000	292,696	7,848,235	164,812,931
Totals KB		379,829,064	9,006,384	14,093,936	402,929,384

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_misc_fg	6	17,300	103,800	98,098	99,877
mssql70_cs_fg	6	33,100	198,600	198,219	198,219
mssql70_ord_fg	6	21,000	126,000	97,168	122,351
Total Allocated MB			**428,442**	**393,528**	**420,489**

	MB	
Dynamic Space MB	102,014	Sum of data for orders, order_line & history
Static Space	291,472	Sum of data+index+5% - Dynamic Space
Free Space	34,956	Total allocated space - (Dynamic & Static Spaces)
Daily Growth	20,466	(Dynamic Space / (W * 62.5)) * tpmC
Daily Spread	4,257	Free space - 1.5 * Daily growth (zero if negative)
	0	SQL Server can be configured to eliminate Daily Spread
180 Day Space MB	3,975,392	Static Space + 180 * (Daily Growth + Daily Spread)
180 Day Space GB	**3,882.22**	
8 hr log GB	**147.16**	(need double for mirroring)
Disk Capacity MB	34728	**33.914 GB** Capacity of 36GB disks (10,000 RPM)
	17408	**17.000 GB** Capacity of 18GB disks (15,000 RPM)

Space Usage	**GB Needed**	**Disks Priced**	**GB Priced**	
180-day space DB	3882.22 GB	252	4284.00 GB	18GB drives
		0	0.00 GB	36GB drives
Total DB		**252**	4284.00 GB	
8-hr log+mirror	294.31 GB	10	339.14 GB	36GB drives
OS, SQL Server	3.10 GB	1	17.00 GB	18GB drive
Total space	**4179.63 GB**	**263**	**4640.14 GB**	

TPC-C 180-Day Dynamic Table Growth Rates for 8 Hours						**61,390.43**	**tpmC**
Tables	**Initial (KB)**	**Final (KB)**	**Change(KB)**	**Unused (KB)**	**KB / New-Order**	**8-Hr MB**	
History	8,189,632	8,879,096	689,464	11,976	0.0618	9,776.41	
Orders	6,772,872	8,300,808	1,527,936	4,384	0.1370	10,556.00	
Order_line	91,994,216	99,840,160	7,845,944	6,864	0.7034	110,079.56	
Dynamic	**106,956,720**	**117,020,064**	**10,063,344**	**23,224**	**0.9022**	**130,411.97**	
New_order	698,256	1,098,976	400,720	45,648	0.0359	1,715.69	
Static							
Log	4,523,621	61,220,303	56,696,682		**5.0829**	150,687.27	147.156
SUM(d_next_o_id)	146,928,960	158,083,337	11,154,377				

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

January 15, 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 benchmark testing.

All pricing shown is in US Dollars ($).

Part Number	Description	Unit Price	Quantity	Price
810-00846	**SQL Server 2000 Enterprise Edition** *Per processor licensing* *Discount schedule: Open Program Level C*	$ 16,541	8	$ 132,328
C11-00821	**Windows 2000 Server** *Server license only - No CALs* *Discount schedule: Open Program - No Level*	$ 738	1	$ 738
048-00317	**Visual C++ Professional 6.0 Win32**	$ 549	1	$ 549
	5-year maintenance for above software	$ 2,095	1	$ 10,475

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.

COMARK TECHNOLOGY SERVICES

Advanced Systems Division

Quote # DM010115-UN004

Jan 15, 2001

To: Unisys Corporation TPC-C

Qty	Prod #	Description	Unit Price	Net Price
ES5085R Server				
1	ESR508152-GZN	SYS: ES5085R Server, 0CPU, 0MB	$13,407.00	**$13,407.00**
8	XEO37002-2MB	PROC: 700MHz Pentium III Xeon /2MB Cache	$3,315.00	**$26,520.00**
32	DIM6168-1GB	MEM: 1GB Memory, SDRAM, Buf 6ns	$3,094.00	**$99,008.00**
1	ESR82-MEZ	BRD: Processor Mezzanine Board, 0 Proc.	$958.00	**$958.00**
1	ESR81-MCB	BRD: Memory Carrier Board, 0 Mem.	$737.00	**$737.00**
2	ESR81-CC4	MEM: Cache Coherency Filter, 4x SRAM	$774.00	**$1,548.00**
7	RAD6004-P64	CTRL: RAID, PCI, 4-Ch w/ 0MB Mem	$1,400.00	**$9,800.00**
7	RAD6324-MEB	MEM: RAID 32MB Cache & Bat. BU	$479.00	**$3,353.00**
1	PCI10001-CXG	CTRL: cLAN, PCI Host Adapter & Cable	$586.00	**$586.00**
1	HDM18110-CX1	DISK: 18GB, 10K SCSI LVD, SCA	$568.00	**$568.00**
1	WI8003-PGS	MAINT: 3-Yr. Performance-Gold Svc Wrrnty Upgrd	$1,768.00	**$1,768.00**
1	EVG2100-P	MONITOR: 15-inch Color	$221.00	**$221.00**
1	WSD200008-L	O/S: Microsoft Windows 2000 Datacenter Server	$22,100.00	**$22,100.00**
5	DUS200008-L	O/S: Win2K Datacenter Annual Subscription	$6,335.00	**$31,675.00**
1	ESS508020-N	SYS MGT: ES7000 Value Add Software	$368.00	**$368.00**
Storage				
278	ESM18308-W45	DISK: 18GB Drive, 15K SCSI LVD, SCA	$737.00	**$204,886.00**
12	OSD36209-W45	DISK: 36GB Drive, 10K SCSI LVD, SCA	$1,172.00	**$14,064.00**
36	ESM310300-L05	CAB: Disk, 8 SCA w/ I/F cards, 0 Disks, 3U	$2,100.00	**$75,600.00**
2	ESM311000-LR	CAB: Disk, 8 SCA w/ RAID Cntl'r, 0MB, 0 Disks, 3U	$5,166.00	**$10,332.00**
2	OSM1032-MEM	MEM: 32MB OSM cache	$171.00	**$342.00**
2	OSM3000-BPF	PWR: 2nd Power Supply Upgrade, OSM	$530.00	**$1,060.00**
1	UPD30001-SXR	PWR:3000 VA UPS, 3U	$1,897.00	**$1,897.00**
8	SFR9-PWR	PWR: Distribution Strip, 220V	$111.00	**$888.00**
20	CBL134-5	CBL: SCSI 68-pin VHD Conn's, 5 meter	$165.00	**$3,300.00**
18	CBL134-CAT	CBL: SCSI 68-pin VHD Conn's, 0.5 meter	$73.00	**$1,314.00**
38	OSM3000-RMK	CAB: Rackmount Kit for Disk Cages	$131.00	**$4,978.00**
4	RM361934-OFE	CAB: 36U x 19" x 34" Cabinet, Open	$810.00	**$3,240.00**
4	RM3619-RDR	DOOR: 36U x 19", Rear	$277.00	**$1,108.00**
4	RM3634-SDS	PNL: 36U x 34" Side Skins, L&R	$221.00	**$884.00**
ES2024 Servers				
3	ES202141-GZN	SYS: ES2024 Tower, w/ 0 Proc., 0MB Mem	$1,547.00	**$4,641.00**
6	CPU3866133-256	PROC:1x866MHz Pentium III/256KB Cache	$368.00	**$2,208.00**
3	VRM3-83	ACC: Voltage Regulator	$29.00	**$87.00**
12	DIM13368-256	MEM: 256 MB SDRAM PC133ECC Memory	$368.00	**$4,416.00**
3	ES2024-18G	DISK: 18GB SCSI 3.5 Internal	$553.00	**$1,659.00**
3	PCI10001-CXG	CTRL: cLAN, PCI Host Adapter & Cable	$586.00	**$1,758.00**
3	WI2003-PGS	MAINT: 3 Yr. Performance-Gold Svc Warranty Upgrade	$1,031.00	**$3,093.00**
3	EVG2100-P	MONITOR:15-inch Color	$221.00	**$663.00**
Network				
1	SWT50082-CXG	SWITCH: cLAN, 8-port, 1.25Gbit	$5,157.00	**$5,157.00**
3	ETH410-T16	SWITCH: Ethernet, 16-Port 10/100TX Fast	$884.00	**$2,652.00**

COMARK
TECHNOLOGY SERVICES
Advanced Systems Division

Quote # DM010115-UN004
Jan 15, 2001

To: Unisys Corporation TPC-C

Qty	Prod #	Description	Unit Price	Net Price
1		Large Purchase Cash Prepay Discount	-$56,284.00	**-$56,284.00**

Prices may vary when items are purchased separately.
Disks come with a 5 year return-to-factory warranty,
7 day replenishment. Quote valid for 90 days.

TOTAL **$506,560.00**

ARK PC Technology, Inc.
dba ARK TECHNOLOGIES
1607 W. Orange Grove Ave., Suite A&B
Orange, CA 92868-1116
Tel: (714) 997-4597 • Fax: (714) 997-4596
www.arkpc.com

January 15, 2001

Attn: Glenn Weeks
Unisys Corporation
25725 Jeronimo Road
Mission Viejo, CA 92691
Tel: 949-380-5785
Fax: 949-465-2552

Ref: 010103

Dear Glenn:

Thank you for your interest in ARK networking product. The following is the pricing for the product that you interested. Please review.

Model No.	Description	1-9	10-199	200-999	1K+
CT1017D1	17-port 10Mbps Desktop Hub, 16*RJ-45+1*BNC, 120V AC/DC Adapter	$40.00	$33.50	$32.00	$31.00

Terms & Conditions:
1. Pricing is FOB Origin, Orange, California.
2. Lead Time is 45 days from receipt of purchase order.
3. Pricing valid for 90 days.
4. Credit term is pending.
5. Pricing subject to quantity indicated.
6. ARK offers lifetime limited warranty.

If you have any questions concerning the product or pricing, please do not hesitate to contact me at 714-997-4597 or email me at stedson@arkpc.com. Thank you.

Sincerely,

Stedson Lim
Director of Sales & Marketing

JAN-15-01 12:07 PM ARK TECH. P.02

NETLUX

14180 Live Oak Ave., Unit E
Baldwin Park, Ca. 91760

1-800-789-1780
Phone#626-851-9737
Fax #626-851-9837

January 15, 2001

Glenn Weeks
Unisys Corporation
25725 Jeronimo Road
Mission Viejo, CA 92691
Fax: (949) 465-2552

Quotation

Quantity	Part No.	Description	Unit Price
1-50	DSS-8+	D-Link DSS-8+ 8-port 10/100Mbps FAST Ethernet Switch	$68.95

Terms and Conditions:
FOB Origin
Quote Valid for 90 days
5 Year Warranty

Sincerely,
Martin Parry
NETLUX