
Itautec Philco S.A.

TPC Benchmark™ C
Full Disclosure Report
for
Servidor Itautec MX200 1P
using
Microsoft SQL Server 2000
Enterprise Edition SP3
and
Microsoft Windows Server 2003
Enterprise Edition

First Edition
Submitted June 28, 2005

First Edition - June 28, 2005

Itautec Philco S.A. believes that the information included in this document is accurate as of the publication date. The information in this document is subject to change without notice. Furthermore, Itautec Philco S.A. is not responsible for any errors contained within this document. The pricing information given in this FDR is accurate as of the publication date, June 28, 2005, but Itautec Philco S.A. cannot guarantee that all sources will offer the same pricing.

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

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

Servidor Itautec is a registered trademark of Itautec Philco S.A.

Microsoft, Windows 2003 and SQL Server 2000 are either trademarks or registered trademarks of Microsoft Corporation.

Intel and Xeon are registered trademarks of Intel Corporation.

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

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

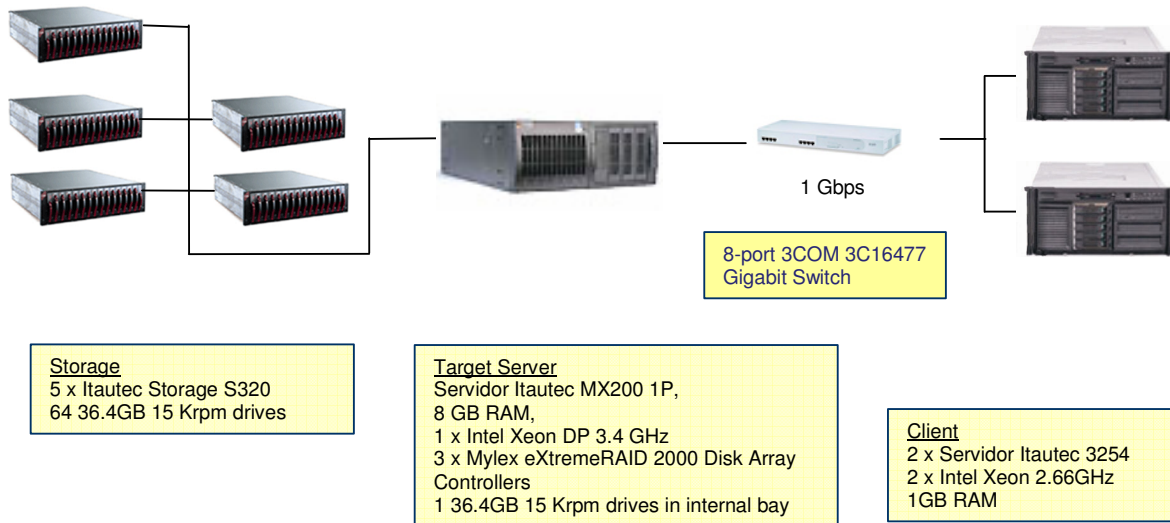


**Servidor Itautec MX200 1P
C/S with Servidor Itautec
3254**

TPC-C Rev. 5.4

Report Date: 06/28/2005

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date
R\$ 465.154 BRL	31,253	R\$ 14,88 BRL	06/28/2005
Processors	Database Manager	Operating System	Other Software
1 Intel Xeon DP 3.4 GHz – Server 2 Intel Xeon 2.66 GHz – Client	Microsoft SQL Server 2000 Enterprise Edition SP3	Windows Server 2003 Enterprise Edition	Microsoft Visual C++ Microsoft COM+
Number of Users			
25,000			



System Components	Server		Each Client	
	Quantity	Description	Quantity	Description
Processor	1	3.4 GHz Intel Xeon DP w/ 2 MB cache / 533 MHz frontside BUS	2	2.66 GHz Xeon w/ 512 KB cache / 533 MHz frontside BUS
Memory	8	1 GB Register DDR ECC 266MHz	2	512 MB Register SDRAM ECC DIMM
Disk Controllers	3	Mylex eXtremeRAID 2000	1	Adaptec AIC-7899 Ultra2
Disk Drives	1	36.4 GB SCSI Drive 15 Krpm	1	36.4 GB 10Krpm SCSI Drive
	64	36.4 GB SCSI Drive 15 Krpm		
Total Storage	2366 GB		36.4 GB	
Tape Drives	1	20/40 GB DAT		

Numerical Quantities Summary

MQTh, Computed Maximum Qualified Throughput

31,253 tpmC

Response Times (in seconds)

	Average	90 th	Max
New Order	0.43	0.68	5.27
Payment	0.26	0.46	4.92
Order Status	0.37	0.60	5.27
Delivery (interactive)	0.10	0.11	0.98
Delivery (deferred)	0.50	0.96	2.77
Stock Level	2.31	3.41	9.75
Menu	0.10	0.11	1.41

Response time delay added for emulated components

Menu 0.1
Resp. 0.1

Transaction Mix, in percent of total transactions


New-Order	44.94%
Payment	43.03%
Order-Status	4.01%
Delivery	4.01%
Stock-Level	4.01%

Keying/Think Times (in seconds)

	Min		Average		Max	
New Order	18.00	0.00	18.01	12.06	18.04	120.51
Payment	3.00	0.00	3.01	12.04	3.04	120.51
Delivery	2.00	0.00	2.01	5.06	2.03	50.50
Stock Level	2.00	0.00	2.01	5.06	2.03	50.50
Order Status	2.00	0.00	2.01	10.06	2.04	100.50

Test Duration

Ramp-up time	43 minutes
Measurement interval	120 minutes
Number of checkpoints	4
Checkpoint interval	30 minutes
Number of transactions (all types) completed in measurement interval	8,680,122

 Itautec Philco	Servidor Itautec MX200 1P			TPC-C Rev. 5.4		
				Report Date: June 28, 2005		
Description	Part Number	Third Party Brand	Unit Price Pricing R\$	Quantity	Extended Price	3 yr. Maint. Price
Server Hardware						
Servidor Itautec 3040 MX200 1P						
Base System with 1 x Xeon (nocona) DP 3.4Ghz/2M	C03SY	Itautec	1	11.130	1	11.130
1 GB MEM REGSDECCDDR2 400 1GBLPN(SB)	H4489	Itautec	1	1.809	8	14.471
36GB Ultra320 15K rpm	F4637	Itautec	1	2.209	1	2.209
On-Board Intel PRO1000XT LAN	Included	Itautec	1	-	1	-
On-Board Intel PRO100+ LAN	Included	Itautec	1	-	1	-
CD-ROM, Internal SCSI Adapter	Included	Itautec	1	-	1	-
Upgrade to 3-year / 4-hour response / 7days - 24hrs	2106_05	Itautec	1	451	1	451
Monitor 15" SYNC MASTER 551S	E9801	Itautec	1	410	1	410
UPS SMART APC 1	92759	Itautec	1	3.469	1	3.469
					Subtotal	31.689
						451
Disk Subsystem						
Extreme RAID 2000 4 Channel Controller	A4805	Itautec	1	10.323	3	30.969
Itautec Storage Ultrab SC2100CTR-AC	F1390	Itautec	1	20.283	5	101.415
36GB Ultra320 15K rpm	F6853	Itautec	1	2.728	64	174.592
36GB Ultra320 15K rpm (10% spare)	F6853	Itautec	1	2.728	7	-
EXT SCSI Cable Eurou VHDCI68P	E8158	Itautec	1	692	5	3.460
Kit Rack for Storage	F1781	Itautec	1	345	5	1.725
Rack Itautec F2400	F2400	Itautec	1	5.281	1	5.281
					Subtotal	317.442
						19.096
Server Software						
Microsoft Windows 2003 Enterprise Edition	G1178	Microsoft	1	15.067	1	15.067
SQL SVR 2000 ENT/1 PROCES LICENSE CD ING	H0138	Microsoft	1	75.355	1	75.355
					Subtotal	90.422
						-
Client Hardware						
Servidor Itautec 3254						
Base System with 1 x Xeon DP 2,66GHz	B7PLV	Itautec	1	13.747	2	27.495
1 x Xeon 2,66Ghz BTO Option	F8273	Itautec	1	1.821	2	3.642
2 x 512MB Memory	F9283	Itautec	1	680	2	1.360
1 x 36GB 10K rpm	F3656	Itautec	1	810	2	1.620
Intel 1000 Base TX Ethernet Controller	Included	Itautec	1	-	2	-
CD-ROM, On-Board LAN	Included	Itautec	1	-	2	-
Upgrade to 3-year / 4-hour response / 7days - 24hrs	2803_04	Itautec	1	532	2	-
Monitor 15" SYNC MASTER 551S	E9801	Itautec	1	410	2	820
					Subtotal	34.936
						1.063
Client Software						
Microsoft Windows 2000 Server	C11-00821	Microsoft	1	3.956	2	7.911
Visual C++ .NET 2003 CD Ing	254-00170	Microsoft	1	423	1	423
					Subtotal	8.334
						-
Network components						
SWITCH 8P BASELINE 10/100/1000	3C16477-ME	Itautec	1	2.148	1	2.148
SWITCH 8P BASELINE 10/100/1000 (2 spares)	3C16477-ME	Itautec	1	2.148	2	-
					Subtotal	2.148
						4.296
					TOTAL	484.971
Large volume purchase with cash in advance Discount on Itautec Hardware (11.58%)						44.724
					TOTAL	440.247
						24.906
Pricing: 1 - Itautec					3-Yr. Cost of Ownership: R\$ 465.154 BRL	
					tpmC Rating: 31253	
					R\$ / tpmC: R\$ 14,88 BRL	
Audited By Lorna Livingtree of Performance Metrics Inc.						

Prices used in TPC benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing sections of the TPC benchmark specifications. If you find that the stated prices are not available according to these items, please inform the TPC at pricing@tpc.org.

Table of Contents

NUMERICAL QUANTITIES SUMMARY	4
TABLE OF CONTENTS	6
ABSTRACT	8
OVERVIEW	8
AUDITOR	8
PREFACE	9
INTRODUCTION	9
GENERAL ITEMS	10
BENCHMARK SPONSOR	10
APPLICATION CODE AND DEFINITION STATEMENTS	10
PARAMETER SETTINGS	10
CONFIGURATION DIAGRAMS	10
MEASURED CONFIGURATION	11
PRICED CONFIGURATION	11
CLAUSE 1 - LOGICAL DATABASE DESIGN RELATED ITEMS	12
TABLE DEFINITIONS	12
PHYSICAL ORGANIZATION OF THE DATABASE	12
INSERT AND DELETE OPERATIONS	12
HORIZONTAL OR VERTICAL PARTITIONING	12
REPLICATION	12
TABLE ATTRIBUTES	12
CLAUSE 2 - TRANSACTION AND TERMINAL PROFILES RELATED ITEMS	13
RANDOM NUMBER GENERATION	13
SCREEN LAYOUT	13
TERMINAL VERIFICATION	13
INTELLIGENT TERMINALS	13
TRANSACTION PROFILES	13
TRANSACTION MIX	14
DEFERRED DELIVERY MECHANISM	14
CLAUSE 3 - TRANSACTION AND SYSTEM PROPERTIES RELATED ITEMS	15
ATOMICITY	15
COMPLETED TRANSACTION	15
ABORTED TRANSACTION	15
CONSISTENCY	15
ISOLATION	15
DURABILITY	15
LOSS OF DATA / LOSS OF LOG	15
INSTANTANEOUS INTERRUPTION AND LOSS OF MEMORY	16

<u>CLAUSE 4: SCALING AND DATABASE POPULATION RELATED ITEMS</u>	17
CARDINALITY OF THE TABLES	17
DISTRIBUTION OF DATABASE TABLES AND LOGS	17
DATABASE MODEL	17
MAPPING PARTITIONS/REPLICATION	18
60-DAY SPACE	18
<u>CLAUSE 5: PERFORMANCE METRICS AND RESPONSE TIME RELATED ITEMS</u>	19
MEASURED TPMC	19
RESPONSE TIMES	19
KEYING AND THINK TIMES	19
RESPONSE TIME DISTRIBUTION CURVES	20
NEW ORDER RESPONSE TIME VS. THROUGHPUT PERFORMANCE	22
NEW ORDER THINK TIME DISTRIBUTION	23
NEW ORDER THROUGHPUT VS. ELAPSED TIME	23
STEADY STATE METHODOLOGY	24
WORK PERFORMED DURING STEADY STATE	24
MEASUREMENT PERIOD DURATION AND CHECKPOINT DURATION	24
REGULATION OF TRANSACTION MIX	24
TRANSACTION STATISTICS	24
CHECKPOINT COUNT AND LOCATION	24
<u>CLAUSE 6: SUT, DRIVER, AND COMMUNICATION DEFINITION RELATED ITEMS</u>	26
RTE PARAMETERS	26
LOST TERMINAL CONNECTIONS	26
EMULATED COMPONENTS	26
BENCHMARKED AND TARGETED SYSTEM CONFIGURATION DIAGRAMS	26
NETWORK CONFIGURATION	26
NETWORK BANDWIDTH	26
OPERATOR INTERVENTION	26
<u>CLAUSE 7 - PRICING RELATED ITEMS</u>	27
HARDWARE AND SOFTWARE LIST	27
AVAILABILITY DATE	27
MEASURED TPMC	27
COUNTRY SPECIFIC PRICING	27
USAGE PRICING	27
SYSTEM PRICING	28
<u>CLAUSE 9 - AUDIT RELATED ITEMS</u>	29
<u>APPENDIX A – SOURCE CODE</u>	32
<u>APPENDIX B: DATABASE DESIGN</u>	90
<u>APPENDIX C: TUNABLE PARAMETERS</u>	122
<u>APPENDIX D – 60-DAY SPACE</u>	160

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark C test conducted on the Servidor Itautec MX200 1P. The tests were conducted by Centro de Informática / UFPE in Brazil. The tests were run in a client/server configuration using two Servidor Itautec 3040HU as clients. The operating system used for the benchmark was Microsoft Windows Server 2003 Enterprise Edition for server and Microsoft Windows 2000 Server Standard Edition for clients. The database was Microsoft SQL Server 2000 SP3.

All tests were done in compliance with Revision 5.4 of the Transaction Processing Council's TPC Benchmark C Standard Specification. Two standard TPC Benchmark™ C metrics, transactions per minute (tpmC) and price per tpmC (R\$/tpmC) are reported and referred to in this document. The results from the tests are summarized below.

Hardware	Software	Total System Cost	tpmC	R\$/tpmC	Total Solution Availability Date
Servidor Itautec MX200 1P	Microsoft SQL Server 2000 Enterprise Edition Microsoft Windows Server 2003 Enterprise Edition	R\$ 465,154 BRL	31,253	R\$ 14.88 BRL	June 28, 2005

Auditor

The benchmark configuration, methodology, and results were audited by Tom Sawyer of Performance Metrics, Inc. to validate compliance with the TPC specifications.

Preface

The Transaction Processing Performance Council (TPC) developed The TPC Benchmark™ C. The TPC was founded to define transaction processing benchmarks and to disseminate objective, verifiable performance data to the industry. This full disclosure report is based on the TPC Benchmark C Standard Specification Version 5.4.

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

Introduction

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.

Although these specifications express implementation in terms of a relational data model with conventional locking scheme, the database may be implemented using any commercially available database management system (DBMS), database server, file system, or other data repository that provides a functionally equivalent implementation. The terms "table", "row", and "column" are used in this document only as examples of logical data structures.

TPC-C uses terminology and metrics that are similar to other benchmarks, originated by the TPC or others. Such similarity in terminology does not in any way imply that TPC-C results are comparable to other benchmarks. The only benchmark results comparable to TPC-C are other TPC-C results conformant with the same revision.

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

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

General Items

Benchmark Sponsor

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

Itautec Philco S.A. was the benchmark sponsor for this TPC Benchmark™ C. The benchmark was developed and engineered by Centro de Informática / UFPE in Brazil.

Application Code and Definition Statements

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

Appendix A contains all source code implemented in this benchmark.

Parameter Settings

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

- *Database tuning options.*
- *Recovery/commit options.*
- *Consistency/locking options.*
- *Operating system and application configuration parameters.*

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

Configuration Diagrams

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

The following pages contain the diagrams for both the tested and priced configurations.

Measured Configuration

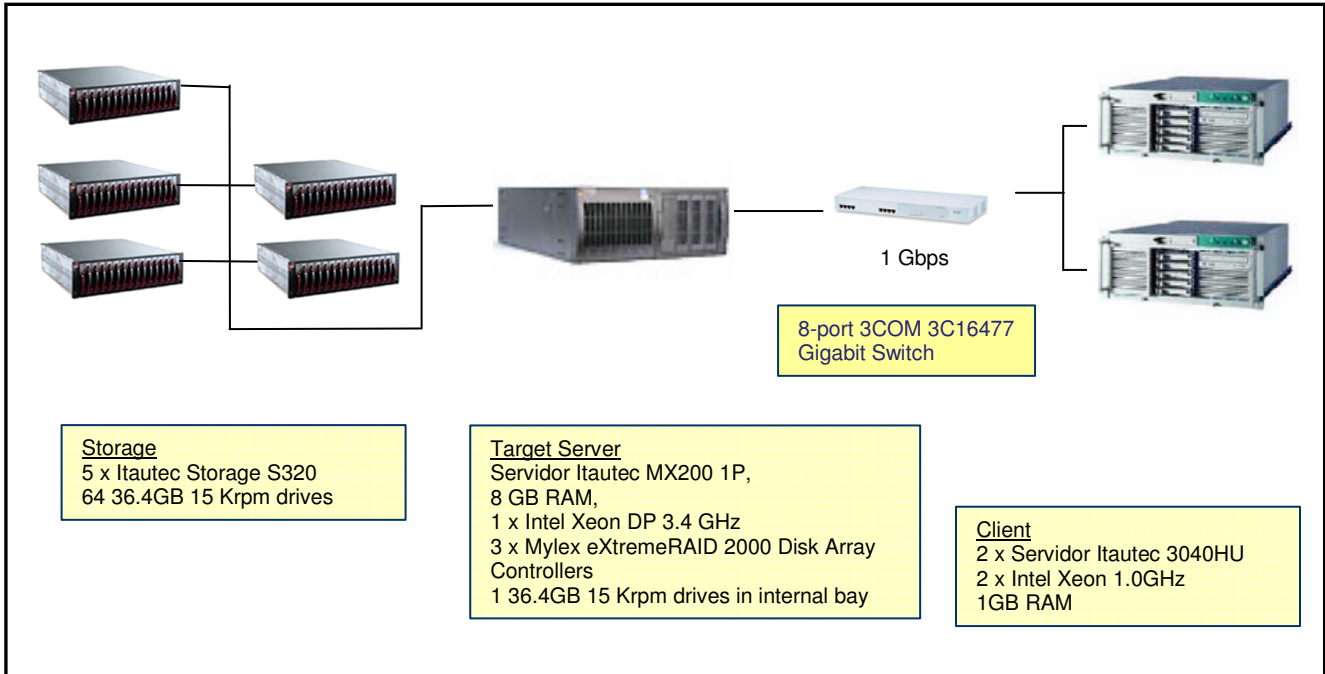


Figure 0.1 – Measured Configuration

Priced Configuration

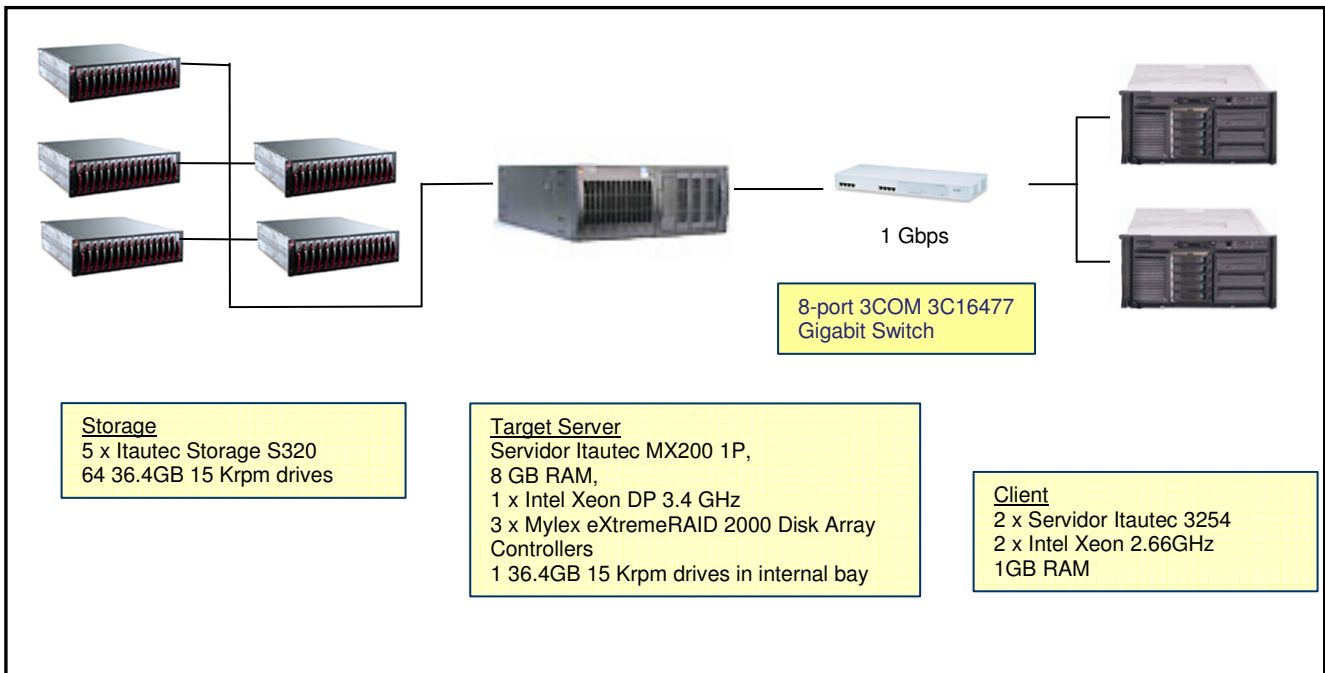


Figure 0.2 – Priced Configuration

Clause 1 - Logical Database Design Related Items

Table Definitions

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

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

Physical Organization of the Database

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

The tested database configuration used 64 disk drives. The physical organization is documented in Table 4.2: Data Distribution.

Insert and Delete Operations

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

Insert and delete functions were fully operational during the running of the benchmark.

Horizontal or Vertical Partitioning

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

Partitioning was not used in this benchmark.

Replication

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

Replication was not used in this benchmark.

Table Attributes

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

No additional attributes were used in this benchmark.

Clause 2 - Transaction and Terminal Profiles Related Items

Random Number Generation

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

The random number generation was handled internally in the Microsoft BenchCraft RTE program. The independent auditing process verified this.

Screen Layout

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

All screen layouts followed the Standard Specifications.

Terminal Verification

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

The auditor with a thorough execution of the five transaction types, using Microsoft Internet Explorer, verified the terminal features.

Intelligent Terminals

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

Comment 1: *The intent of this clause is to describe any special manipulations performed by a local terminal or workstation to off-load work from the SUT. This includes, but is not limited to: screen presentations; message bundling, and local storage of TPC-C rows.*

Comment 2: *This disclosure also requires that all data manipulation functions performed by the local terminal to provide navigational aids for transaction(s) must also be described. Within this disclosure, the purpose of such additional function(s) must be explained.*

The application code responsible for processing the data was executed on the clients. HTML Screen manipulation commands were downloaded to the web browser, which controlled the input and output graphics. This code is documented in Appendix A. IIS (Microsoft Internet Information Server) was involved in processing and presenting this data.

Transaction Profiles

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

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

The number of items per orders 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.

Table 2.1: Transaction Statistics

Transaction	Function	Value
New Order	Home Warehouse Order Lines	99.00%
	Remote Warehouse Order Lines	1.00%
	Rolled Back Transactions	0.99%
	Average Lines Per Order	10.00
Payment	Home Warehouse Transactions	85%
	Remote Warehouse Transactions	15%
	Non-Primary Key Access	60.04%
Order Status	Non-Primary Key Access	60.03%
Delivery	Delivery Transactions Skipped	0

Transaction Mix

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

Table 2.2: Transaction Mix

Transaction	Percentage
New Order	44.94
Payment	43.03
Order Status	4.01
Delivery	4.01
Stock Level	4.01

Deferred Delivery Mechanism

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

Microsoft COM+ components installed on each Web Client create a pool of threads connected to the database, responsible for processing delivery transactions. When the application on the Web Client side receives a request for a delivery transaction, one of the threads is assigned for processing it while the control is immediately returned to the user (RTE). Upon completion, the thread writes an entry in the delivery log and returns to the thread pool.

The source code is listed in Appendix A.

Clause 3 - Transaction and System Properties Related Items

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

All ACID tests were conducted successfully according to specification.

Atomicity

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

Completed Transaction

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

Aborted Transaction

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

Consistency

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

Consistency conditions one through four were tested using a script to issue queries to the database. The results of the queries verified that the database was consistent for all four tests. A run was executed under full load lasting over two hours and including several checkpoints. The script was re-executed and the result of the same queries verified that the database remained consistent after the run.

Isolation

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

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

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

For Isolation test seven, case A was followed.

Durability

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

Loss of Data / Loss of Log

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

- 1) A 2,600 warehouse database was generated.
- 2) The database was backed up using SQL scripts.
- 3) A sum of D_NEXT_O_ID was recorded.
- 4) The RTEs were started with more than 10% of the benchmarked users (25,000).
- 5) The system was run in steady state for 10 minutes.

- 6) One log disk was removed from the drive cabinet. No interruption occurred.
- 7) Keep running more 10 minutes.
- 8) One data disk was removed, causing an SQL Server error.
- 9) The RTE was stopped.
- 10) SQL Server was stopped and restarted.
- 11) The transaction log was dumped to disk.
- 12) SQL Server was stopped, the Windows was shutdown and the machine powered off.
- 13) The failed disk was replaced.
- 14) The machine was powered on.
- 15) SQL Server was started.
- 16) The database and the transaction log were restored from backup.
- 17) The sum of D_NEXT_O_ID was taken.
- 18) This number was compared with the number of new orders reported by the RTE.
- 19) Consistency test #3 was executed and verified.

Instantaneous Interruption and Loss of Memory

To validate system recovery an instantaneous interruption was caused by removing power to the Server, the following steps were executed:

- 1) A sum of D_NEXT_O_ID was taken.
- 2) 25,000 users were started via the RTEs.
- 3) The system was run in steady state for 5 minutes.
- 4) The power supply cord was removed from the server, causing instantaneous interruption.
- 5) The RTE's were stopped.
- 6) Power was reconnected and the system was rebooted.
- 7) SQL Server was started and the recovery process completed successfully.
- 8) A new count of D_NEXT_O_ID was taken.
- 9) This number was validated against the calculated number reported by the RTEs.

Clause 4: Scaling and Database Population Related Items

Cardinality of the Tables

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

The database was generated with 2600 warehouses, and the audited performance run used 2500 warehouses.

Table 4.1: Table Cardinality

Table	Initial Cardinality
Warehouse	2600
District	26000
Customer	78000000
New Order	23400000
Orders	78000000
History	78000000
Order Line	779997753
Item	100000
Stock	260000000
Deleted Warehouses	100

Distribution of Database Tables and Logs

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

The system was configured with 64 36.4GB 15Krpm SCSI disks for the database. There were 56 disks connected to 2 controllers configured as RAID 0 and 8 disks connected to one controller and configured as RAID 0+1. The operating system was stored in one 36.4GB 15Krpm disk. The OS disk was connected to the same controller as the log disks. Most logical data drives contained 2 partitions for miscellaneous, customer and stock, and backup data. Raw file systems were used except for the NTFS formatted backup partitions. The configuration is further detailed below in Table 4.2.

Database Model

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/I, 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 Enterprise Edition is a relational DBMS.

The interface used was Microsoft SQL Server stored procedures accessed with Remote Procedure Calls embedded in C code using the Microsoft ODBC interface.

Table 4.2: Data Distribution

Controller	DB Components	Partition	Size	Disks
0	Miscellaneous Customer and Stock Backup	E:\ G:\ W:\	65,00GB 100,00GB 585,00GB	28 – 36 GB 15K
1	Miscellaneous Customer and Stock Backup	F:\ H:\ X:\	65,00GB 100,00GB 585,00GB	28 – 36 GB 15K
2	Transaction Log	L:\	136,66 GB	8 – 36 GB 15K

Mapping Partitions/Replication

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

No partitioning or replication was used.

60-Day Space

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

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

1. The current log space was recorded by running *dbcc sqlperf(logspace)*
2. Transactions were run against the database with a full user load.
3. The final log space usage was recorded by running *dbcc sqlperf(logspace)*
4. The space used was calculated as the difference between the first and second queries.
5. The number of NEW-ORDERS was retrieved from the RTE report generated for the entire run.
6. The total space used was divided by the number of NEW-ORDERS producing a size per NEW-ORDER.
7. The NEW-ORDER size was multiplied by the measured tpmC rate and multiplied by 480 minutes.

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

The details of the 8-hour transaction log space and the 60-day space requirements are shown in Appendix D.

Clause 5: Performance Metrics and Response Time Related Items

Measured tpmC

Measured tpmC must be reported.

Measured tpmC: **31,253** tpmC

Price per tpmC: **R\$ 14.88 BRL** per tpmC

Response Times

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

Table 5.1: Response Times

Transaction	Average	90%	Maximum
New order	0.43	0.68	5.27
Payment	0.26	0.46	4.92
Order Status	0.37	0.60	5.27
Delivery (interactive)	0.10	0.11	0.98
Delivery (deferred)	0.50	0.96	2.77
Stock Level	2.31	3.41	9.75
Menu	0.10	0.11	1.41

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	Average	Minimum	Maximum
New Order	18.01	18.00	18.04
Payment	3.01	3.00	3.04
Delivery	2.01	2.00	2.03
Stock Level	2.01	2.00	2.03
Order Status	2.01	2.00	2.04

Table 5.3: Think Times

Transaction	Average	Minimum	Maximum
New Order	12.06	0.00	120.51
Payment	12.04	0.00	120.51
Delivery	5.06	0.00	50.50
Stock Level	5.06	0.00	50.50
Order Status	10.06	0.00	100.50

Response Time Distribution Curves

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

Figure 5.1 - New-Order Transaction Response Time Distribution

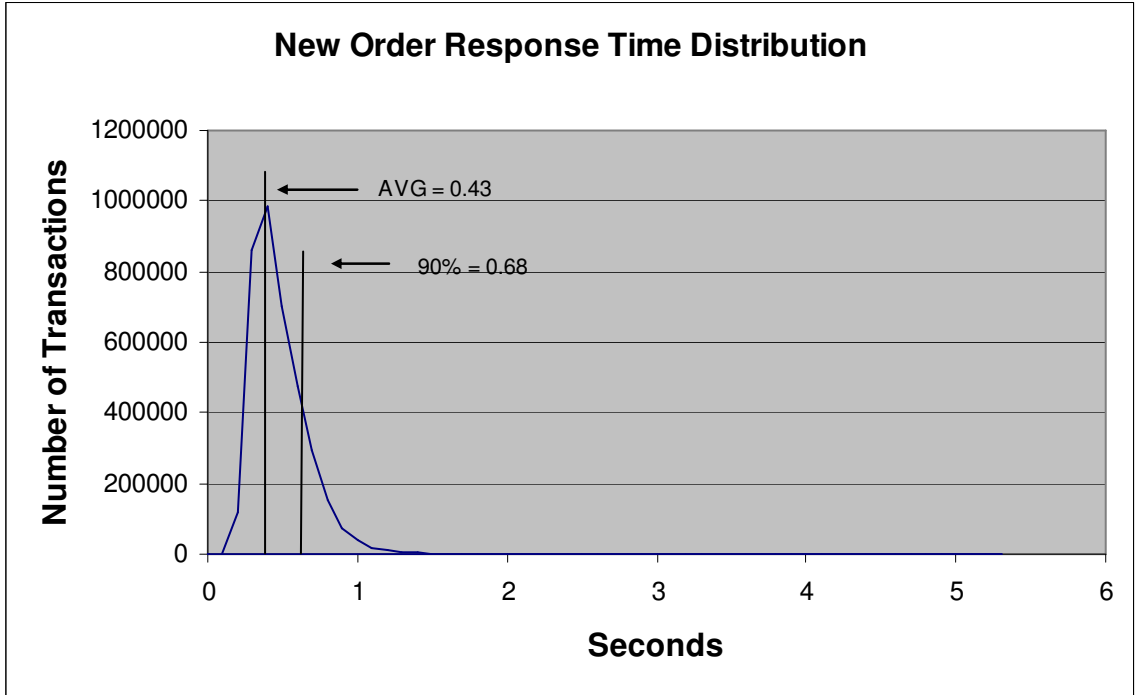


Figure 5.2 - Payment Transaction Response Time Distribution

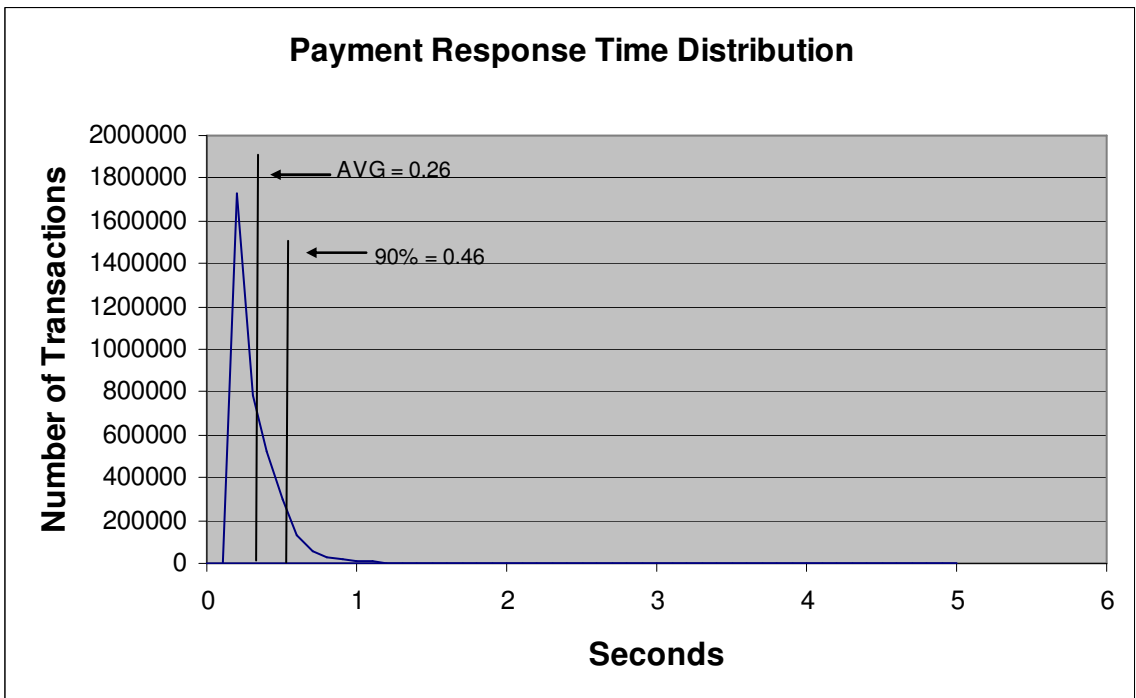


Figure 5.3 – Stock Level Transaction Response Time Distribution

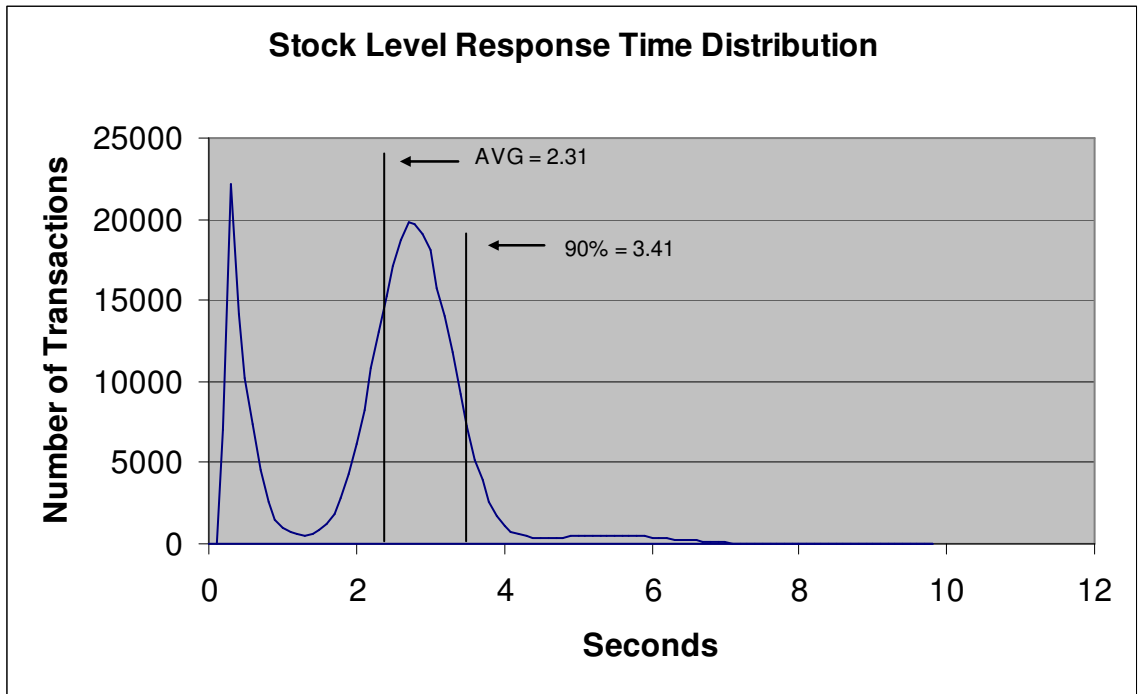


Figure 5.4 – Order Status Transaction Response Time Distribution

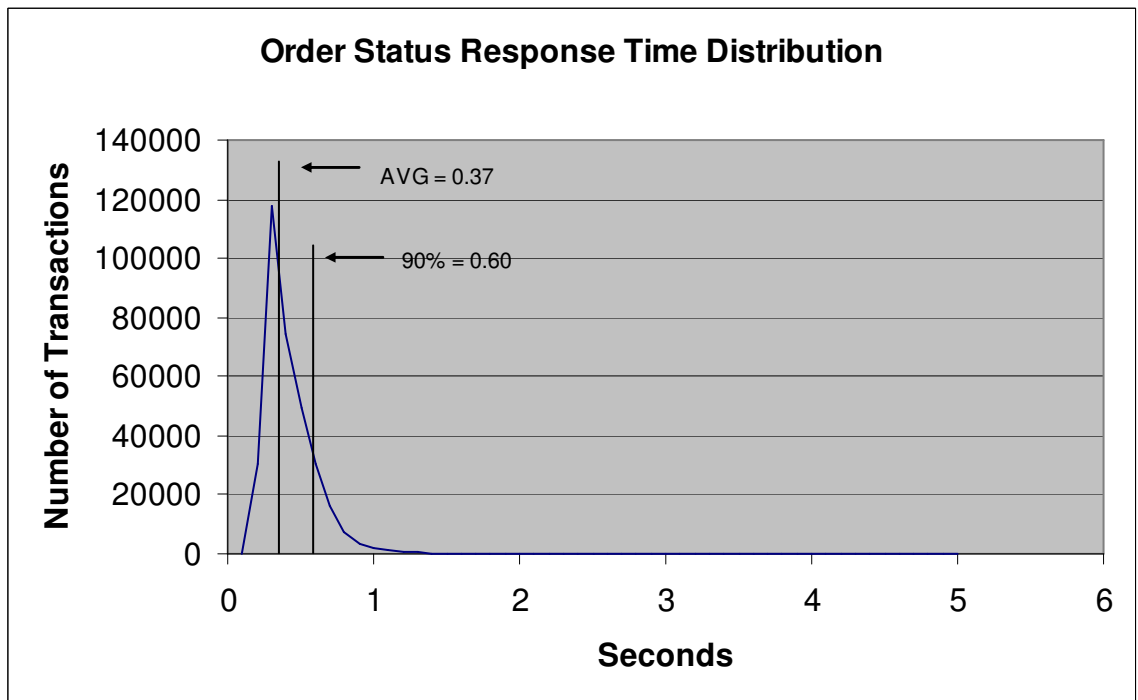


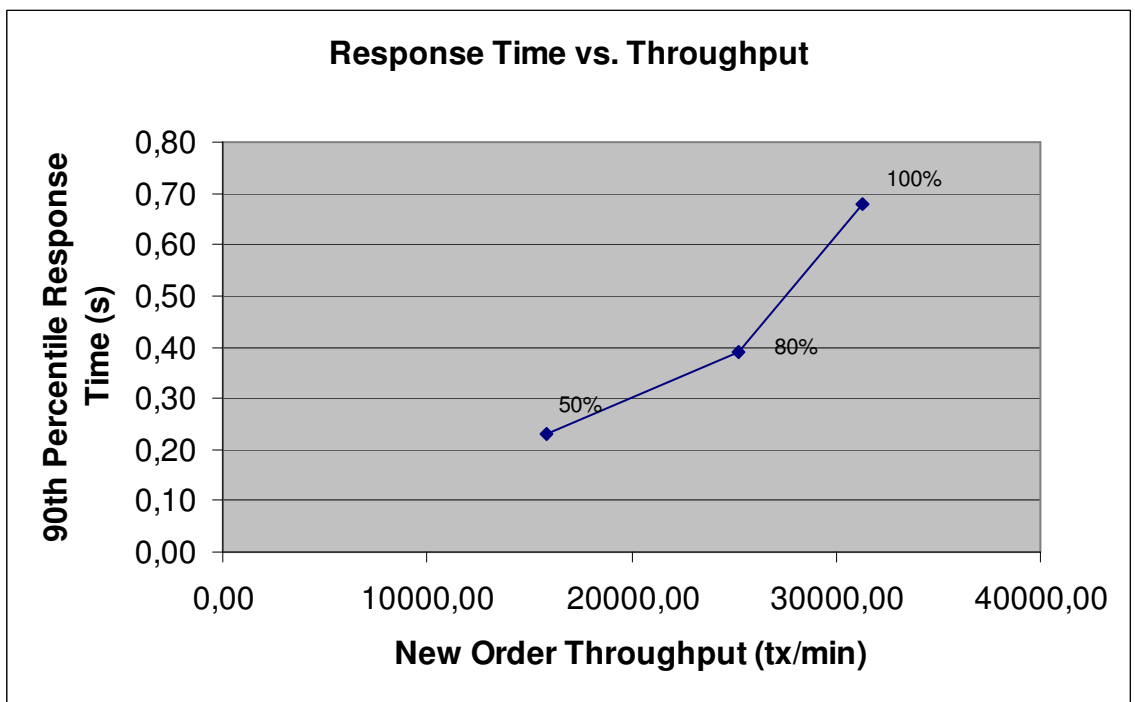
Figure 5.5 – Delivery Transaction Response Time Distribution



New Order Response Time vs. Throughput Performance

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

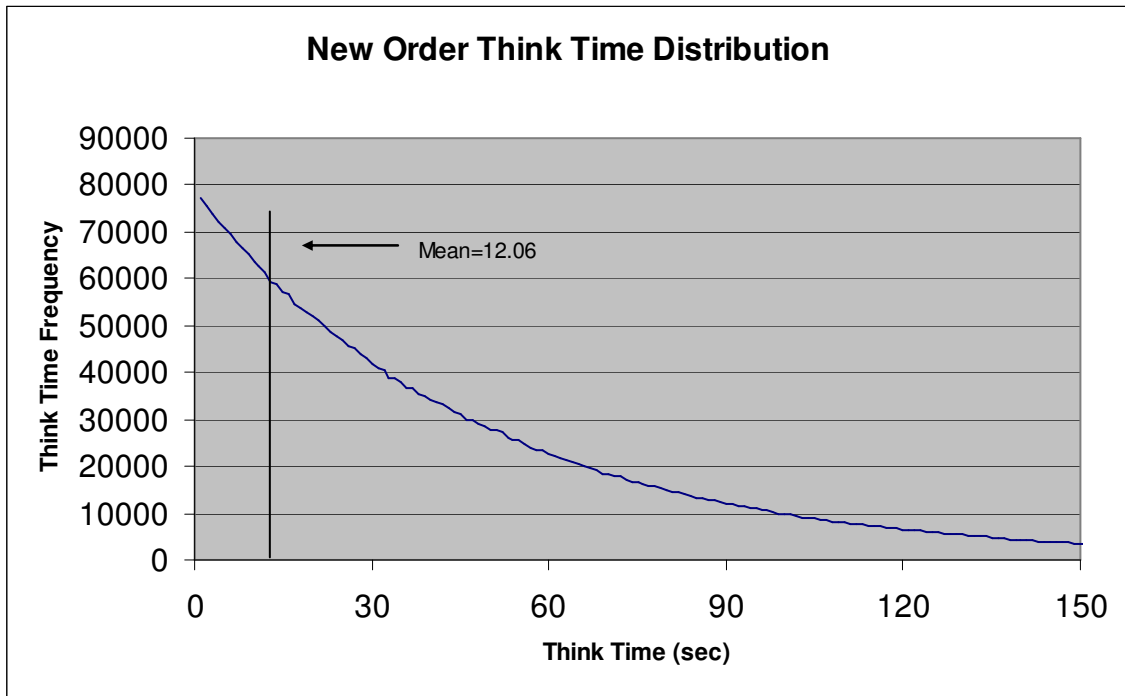
Figure 5.6 – New Order Response Time vs. Throughput



New Order Think Time Distribution

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

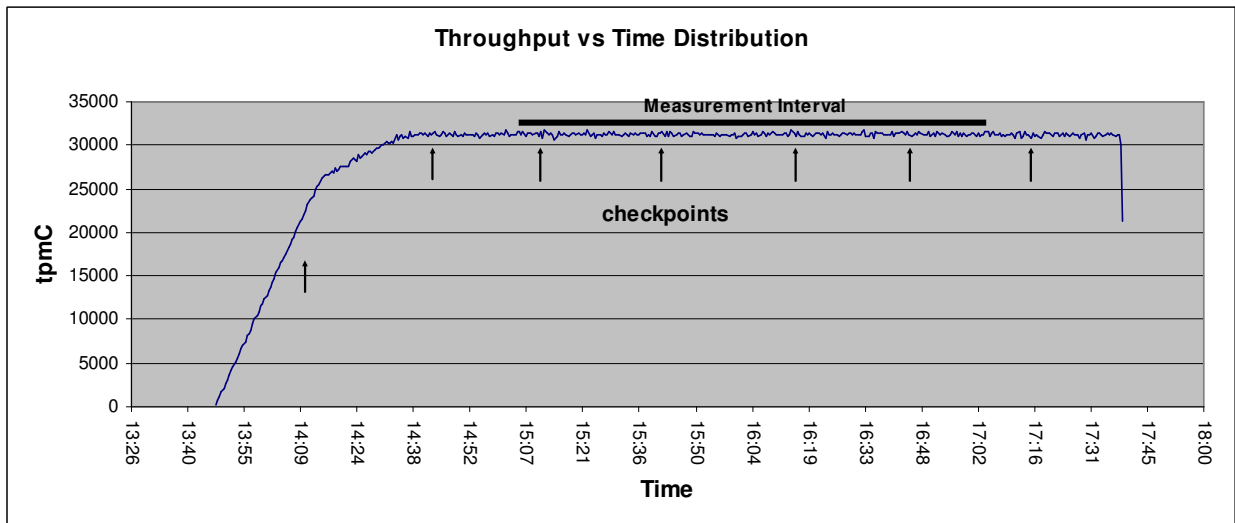
Figure 5.7 – New Order Think Time Distribution



New Order Throughput vs. Elapsed Time

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

Figure 5.8 – Throughput vs. Time Distribution



Steady State Methodology

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.

By using the monitoring tools on the RTE, a steady state was determined. Figure 5.8 further supports the level chosen by the utilities used.

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.

A checkpoint in Microsoft SQL Server writes to disk all updated memory pages that have not been yet actually written to disk. SQL Server recovery interval parameter was set to 115 to perform checkpoint at specific intervals. A checkpoint script, which issues specified number of checkpoint at specified (30 minutes) intervals, was started after all users logged in and sending transactions.

Measurement Period Duration and Checkpoint Duration

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

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

Table 5.9: Checkpoints

	Start	End	Duration (in seconds)
Measurement Interval	15:07:30	17:07:30	7200
1 st Checkpoint	15:13:42	15:34:13	1231
2 nd Checkpoint	15:43:37	16:03:52	1215
3 rd Checkpoint	16:13:32	16:34:19	1247
4 th Checkpoint	16:43:27	17:05:08	1301

Regulation of Transaction Mix

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

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

Transaction Statistics

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

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

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

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

The percentage of remote Payment transactions must be disclosed.

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

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

The above statistics are disclosed in Table 2.1.

Checkpoint Count and Location

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

There was one checkpoint before measurement and four checkpoints during measurement. The time of the first checkpoint during the measurement interval is 1231 seconds from the start of the measurement, and the checkpoint interval is 30 minutes.

Clause 6: SUT, Driver, and Communication Definition Related Items

RTE Parameters

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

Comment: *The intent is to demonstrate the RTE was configured to generate transaction input data as specified in Clause 2.*

The RTE input parameters are listed in Appendix C - Tunable Parameters.

Lost Terminal Connections

The number of terminal connections lost during the Measurement Interval must be disclosed.

No terminal connections were lost.

Emulated Components

It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to that of the priced system. The results of the test described in Clause 6.6.3.4 must be disclosed. (8.1.7.2)

No components were emulated.

Benchmarked and Targeted System Configuration Diagrams

A complete functional diagram of both the benchmark configuration and the configuration of the proposed (target) system must be disclosed. A detailed list of all software and hardware functionality being performed on the Driver System, and its interface to the SUT must be disclosed (see Clause 6.6.3.6). (8.1.7.3)

The driver system performed transaction data generation and communication to the client through the standard Web browser (HTTP) protocol. It also captured and time stamped the SUT output data for post-processing of the reported metrics. No other functionality was included on the driver system.

Figures 0.1 and 0.2 of this report contain detailed diagrams of both the benchmark configuration and the priced configuration.

Network Configuration

The network configurations of both the tested services and the proposed (target) services which are being represented and a thorough explanation of exactly which parts of the proposed configuration are being replaced on the Driver System must be disclosed (see Clause 6.6.4). (8.1.7.4)

The network configurations of the benchmarked and priced configurations were identical.

Network Bandwidth

The bandwidth of the network(s) used in the tested/priced configuration must be disclosed. (8.1.7.5)

The bandwidth of the tested and priced networks were as follows:

- 100 BaseT (100 Mbit/sec) network segments between the RTE and the clients
- 1000 BaseT (1000 Mbit/sec) between the Clients and Server.

Operator Intervention

If the configuration requires operator intervention (see Clause 6.6.6), the mechanism and the frequency of this intervention must be disclosed. (8.1.7.6)

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

Clause 7 - Pricing Related Items

Hardware and Software List

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) of price(s) must also be reported. (8.1.8.1)

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. (8.1.8.2)

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

Availability Date

The committed delivery date for general availability (availability date) of products used in the price calculations must be reported. When the priced system includes products with different availability dates, the reported availability date for the priced system must be the date at which all components are committed to be available. (8.1.8.3)

Hardware Availability Date: 06/28/2005

Software Availability Date: 06/28/2005

Measured TpmC

A statement of the measured tpmC, as well as the respective calculations for 5-year pricing, price/performance (price/tpmC), and the availability date must be included. (8.1.8.4)

Maximum Qualified Throughput: 31,253 tpmC

Price Performance Metric: R\$ 14.88 BRL

Country Specific Pricing

Additional Clause 7 related items may be included in the Full Disclosure Report for each country specific priced configuration. Country specific pricing is subject to Clause 7.1.7. (8.1.8.5)

This system is priced for Brazil.

Usage Pricing

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

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

Comment: *Usage pricing may include, but is not limited to, the operating system and database management software.*

The component pricing based on usage is shown below:

- 2 Microsoft Windows 2000 Server Licenses
- 1 Microsoft Windows Server 2003 Enterprise Edition License
- 1 Microsoft SQL Server 2000 Enterprise Edition Licenses (per processor)
- 1 Microsoft Visual C++ 32 bit Edition
- 3 Year Support for Hardware Components.

System Pricing

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. An example of the standard pricing sheet is shown in Appendix B. (8.1.8.7)

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

The details of the hardware and software are reported in the front of this report as part of the Executive Summary.

Clause 9 - Audit Related Items

Auditor

The auditor's name, address, phone number, and a copy of the auditor's attestation letter indicating compliance must be included in the Full Disclosure Report. (8.1.9.1)

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

This TPC-C benchmark has been audited by Lorna Livingtree of Performance Metrics.

Availability of the Full Disclosure Report

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

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

Transaction Processing Performance Council
c/o Administrator, TPC
Presidio of San Francisco
Bldg 572B Ruger St.
San Francisco, CA 94129-0920
Phone: (415) 561-6272, fax (415) 561-6120
www.tpc.org

or:

Itautec Philco S.A.
Rua Santa Catarina, 1
030860-025 – São Paulo – SP
Phone: +55 (11) 6097-3000, fax +55 (11) 6097-4284

or:

Laboratório de Análise de Performance
Centro de Informática / UFPE
Rua Prof. Luis Freire, S/N – Cidade Universitária
50740-540 – Recife – PE
Phone: +55 (81) 3453-9213

PERFORMANCE METRICS INC.
TPC Certified Auditors



June 27, 2005

Mr. Fábio Ávila Rêgo Pessoa
Itautec Performance Lab
Centro de Informática – UFPE
Recife, Brazil

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

Platform: Servidor Itautec MX200 1P

Database Manager: Microsoft SQL Server 2000 Enterprise Edition 32-bit SP3

Operating System: Microsoft Windows 2003 Server, Enterprise Edition

Transaction Monitor: Microsoft COM+

Server: Servidor Itautec MX200 1P				
CPU's	Memory	Disks	90% Response	TpmC
1 Intel Xeon @ 3.4 GHz	Main: 8 GB	65 @ 36GB	0.68 sec	31,253
Client: Servidor Itautec 3040HU				
2 Intel Xeon @ 1 GHz	Main: 1 GB	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 files were properly sized and populated.
- The database was properly scaled with 2600 warehouses of which only 2500 were active during the measured interval.

PERFORMANCE METRICS INC.
TPC Certified Auditors

- The ACID properties were successfully demonstrated.
- Input data was generated according to the specified percentages.
- Eight hours of mirrored log space was present on the tested system.
- Eight hours of growth space for the dynamic tables was present on the tested system.
- The data for the 60 day space calculation was verified.
- The controller cache was disabled on the log disk controller.
- The steady state portion of the test was 120 minutes.
- At least one checkpoint was taken before the measured interval.
- Four complete checkpoints were taken during the measured interval.
- The system pricing was checked for major components and maintenance.

Auditor Notes:

The client machines used in the tested configuration are no longer orderable from the sponsor. The clients substituted in the priced configuration meet the substitution requirements.

Sincerely,



Lorna Livingtree
Auditor

Appendix A – Source Code

isapi_dll/src/tpcc.def

```
LIBRARY TPCC.DLL

EXPORTS

    GetExtensionVersion @1
    HttpExtensionProc @2
    TerminateExtension @3
```

Isapi_dll/src/tpcc.h

```
/* FILE: TPCC.H Microsoft TPC-C
 * Kit Ver. 4.20.000 Copyright
 * Microsoft, 1999 All Rights Reserved
 * Version
 * 4.10.000 audited by Richard Gimarc, Performance
 * Metrics, 3/17/99
 * PURPOSE: Header file for ISAPI TPCC.DLL,
 * defines structures and functions used in the isapi
 * tpcc.dll.
 */

//VERSION RESOURCE DEFINES
#define _APS_NEXT_RESOURCE_VALUE 101
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101

#define TP_MAX_RETRIES 50

//note that the welcome form must be processed first as
//terminal ids assigned here, once the
//terminal id is assigned then the forms can be
//processed in any order.
#define WELCOME_FORM 1
//beginning form no term id assigned, form id
#define MAIN_MENU_FORM 2
//term id assigned main menu form id
#define NEW_ORDER_FORM 3 //new
order form id
```

```
#define PAYMENT_FORM 4
//payment form id
#define DELIVERY_FORM 5
//delivery form id
#define ORDER_STATUS_FORM 6 //order status
id
#define STOCK_LEVEL_FORM 7 //stock level
form id

//This macro is used to prevent the compiler error
//unused formal parameter
#define UNUSEDPARAM(x) (x = x)

//This structure defines the data necessary to keep
//distinct for each terminal or client connection.
typedef struct _CLIENTDATA
{
    int iNextFree; //index of next
    int w_id; //warehouse id
    int d_id; //district id
    int iSyncId; //synchronization id
    int iTickCount; //time of last
    access;
    CTPCC_BASE *pTxn;
} CLIENTDATA, *PCLIENTDATA;

//This structure is used to define the operational
//interface for terminal id support
typedef struct _TERM
{
    int iNumEntries;
    int iFreeList; //total allocated terminal array entries
    int iFreeList; //next available terminal array element or -1
    int iMastersSyncId; //synchronization id
    CLIENTDATA *pClientData; //pointer to
    allocated client data
} TERM;
typedef TERM *PTERM; //pointer to
terminal structure type

enum WEBERROR
{
    NO_ERR,
    ERR_COMMAND_UNDEFINED,
    ERR_D_ID_INVALID,
```

```
ERR_DELIVERY_CARRIER_ID_RANGE,
ERR_DELIVERY_CARRIER_INVALID,
ERR_DELIVERY_MISSING_OCD_KEY,
ERR_DELIVERY_THREAD_FAILED,
ERR_GETPROCADDR_FAILED,
ERR_HTML_ILL_FORMED,
ERR_INVALID_SYNC_CONNECTION,
ERR_INVALID_TERMID,
ERR_LOADDLL_FAILED,
ERR_MAX_CONNECTIONS_EXCEEDED,
ERR_MEM_ALLOC_FAILED,
ERR_MISSING_REGISTRY_ENTRIES,
ERR_NEWORDER_CUSTOMER_INVALID,
ERR_NEWORDER_CUSTOMER_KEY,
ERR_NEWORDER_DISTRICT_INVALID,
ERR_NEWORDER_FORM_MISSING_DID,
ERR_NEWORDER_ITEMID_INVALID,
ERR_NEWORDER_ITEMID_RANGE,
ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
ERR_NEWORDER_MISSING_ID_KEY,
ERR_NEWORDER_MISSING_QTY_KEY,
ERR_NEWORDER_MISSING_SUPPW_KEY,
ERR_NEWORDER_NOITEMS_ENTERED,
ERR_NEWORDER_QTY_INVALID,
ERR_NEWORDER_QTY_RANGE,
ERR_NEWORDER_QTY_WITHOUT_SUPPW,
ERR_NEWORDER_SUPPW_INVALID,
ERR_NO_SERVER_SPECIFIED,
ERR_ORDERSTATUS_CID_AND_CLT,
ERR_ORDERSTATUS_CID_INVALID,
ERR_ORDERSTATUS_CLT_RANGE,
ERR_ORDERSTATUS_DID_INVALID,
ERR_ORDERSTATUS_MISSING_CID_CLT,
ERR_ORDERSTATUS_MISSING_CID_KEY,
ERR_ORDERSTATUS_MISSING_CLT_KEY,
ERR_ORDERSTATUS_MISSING_DID_KEY,
ERR_PAYMENT_CDI_INVALID,
ERR_PAYMENT_CID_AND_CLT,
ERR_PAYMENT_CUSTOMER_INVALID,
ERR_PAYMENT_CWI_INVALID,
ERR_PAYMENT_DISTRICT_INVALID,
ERR_PAYMENT_HAM_INVALID,
ERR_PAYMENT_HAM_RANGE,
ERR_PAYMENT_LAST_NAME_TO_LONG,
ERR_PAYMENT_MISSING_CDI_KEY,
ERR_PAYMENT_MISSING_CID_CLT,
ERR_PAYMENT_MISSING_CID_KEY,
ERR_PAYMENT_MISSING_CLT,
ERR_PAYMENT_MISSING_CLT_KEY,
ERR_PAYMENT_MISSING_CWI_KEY,
ERR_PAYMENT_MISSING_DID_KEY,
ERR_PAYMENT_MISSING_HAM_KEY,
ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
ERR_STOCKLEVEL_THRESHOLD_INVALID,
ERR_STOCKLEVEL_THRESHOLD_RANGE,
ERR_VERSION_MISMATCH,
ERR_W_ID_INVALID
};

class CWEBCLNT_ERR : public CBaseErr
{
public:
    CWEBCLNT_ERR(WEBERROR Err)
    {
        m_Error = Err;
        m_szTextDetail = NULL;
        m_SystemErr = 0;
        m_szErrorText = NULL;
    };
};
```



```

        CWEBCLNT_ERR(WEBERROR Err, char
*szTextDetail, DWORD dwSystemErr)
    {
        m_Error = Err;
        m_sszTextDetail = new
char[strlen(szTextDetail)+1];
        strcpy( m_sszTextDetail,
szTextDetail );
        m_SystemErr =
dwSystemErr;
        m_sszErrorText = NULL;
    };
    ~CWEBCLNT_ERR()
    {
        if (m_sszTextDetail !=
NULL)
            delete []
m_sszTextDetail;
        if (m_sszErrorText !=
NULL)
            delete []
m_sszErrorText;
    };
    WEBERROR m_Error;
    char
*m_sszTextDetail; //
    char
*m_sszErrorText;
    DWORD
m_SystemErr;

    int ErrorType() {return
ERR_TYPE_WEBDLL;};
    int ErrorNum() {return m_Error;};
    char *ErrorText();
};

//These constants have already been defined in
engstut.h, but since we do
//not want to include it in the delisrv executable
#define TXN_EVENT_START 2
#define TXN_EVENT_STOP 4
#define TXN_EVENT_WARNING 6
//used to record a warning into the log

//function prototypes

BOOL WINAPI DllMain(HANDLE hModule, DWORD
ul_reason_for_call, LPVOID lpReserved);
void WriteMessageToEventLog(LPTSTR lpszMsg);
void ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB,
int *pCmd, int *pFormId, int *pTermId, int *pSyncId);
void WelcomeForm(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer);
void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer);
void BeginCmd(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId, int iTermId);
void ProcessCmd(EXTENSION_CONTROL_BLOCK *pECB, int
iFormId, int iTermId);
void StatsCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer);
void ErrorMessage(EXTENSION_CONTROL_BLOCK *pECB, int
iError, int iErrorType, char *szMsg, int iTermId);
void GetKeyValue(char **pQueryString, char *pkey, char
*pvalue, int iMax, WEBERROR err);
int GetIntKeyValue(char **pQueryString, char *pkey,
WEBERROR NokeyErr, WEBERROR NotIntErr);
void TermInit(void);
void TermDeleteAll(void);
int TermAdd(void);

```

```

void TermDelete(int id);
void ErrorForm(EXTENSION_CONTROL_BLOCK *pECB, int
iType, int iErrorNum, int iTermId, int iSyncId, char
*szErrorText, char *szBuffer );
void MakeMainMenuForm(int iTermId, int iSyncId, char
*szForm);
void MakeStockLevelForm(int iTermId, STOCK_LEVEL_DATA
*pStockLevelData, BOOL bInput, char *szForm);
void MakeNewOrderForm(int iTermId, NEW_ORDER_DATA
*pNewOrderData, BOOL bInput, char *szForm);
void MakePaymentForm(int iTermId, PAYMENT_DATA
*pPaymentData, BOOL bInput, char *szForm);
void MakeOrderStatusForm(int iTermId, ORDER_STATUS_DATA
*pOrderStatusData, BOOL bInput, char *szForm);
void MakeDeliveryForm(int iTermId, DELIVERY_DATA
*pDeliveryData, BOOL bInput, char *szForm);
void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB,
int iTermId, char *szBuffer);
void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB,
int iTermId, char *szBuffer);
void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer);
void GetNewOrderData(LPSTR lpszQueryString,
NEW_ORDER_DATA *pNewOrderData);
void GetPaymentData(LPSTR lpszQueryString, PAYMENT_DATA
*pPaymentData);
void GetOrderStatusData(LPSTR lpszQueryString,
ORDER_STATUS_DATA *pOrderStatusData);
BOOL PostDeliveryInfo(long w_id, short o_carrier_id);
BOOL IsNumeric(char *ptr);
BOOL IsDecimal(char *ptr);
void DeliveryWorkerThread(void *ptr);

```

isapi_dll/src/tpcc.rc

```

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

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

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

#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifdef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // _WIN32

#ifdef _MAC

```

```

////////////////////////////////////
//
// Version
//
VS_VERSION_INFO VERSIONINFO
FILEVERSION 0,4,0,0
PRODUCTVERSION 0,4,0,0
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILESOS 0x40004L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "Comments", "TPC-C HTML DLL Server
(DBLIB)\0"
            VALUE "CompanyName", "Microsoft\0"
            VALUE "FileDescription", "TPC-C HTML DLL
Server (DBLIB)\0"
            VALUE "FileVersion", "0, 4, 0, 0\0"
            VALUE "InternalName", "tpcc\0"
            VALUE "LegalCopyright", "Copyright ©
1997\0"
            VALUE "OriginalFilename", "tpcc.dll\0"
            VALUE "ProductName", "Microsoft tpcc\0"
            VALUE "ProductVersion", "0, 4, 0, 0\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1200
    END
END

#endif // !_MAC

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

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

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

#endif // APSTUDIO_INVOKED

```

```

////////////////////////////////////
//
// Dialog
//
IDD_DIALOG1 DIALOG DISCARDABLE 0, 0, 186, 95
STYLE DS_MODALFRAME | WS_POPUP | WS_CAPTION |
WS_SYSMENU
CAPTION "Dialog"
FONT 8, "MS Sans Serif"
BEGIN
    DEFPUSHBUTTON "OK",IDOK,129,7,50,14
    PUSHBUTTON "Cancel",IDCANCEL,129,24,50,14
END

////////////////////////////////////
//
// DESIGNINFO
//
#ifdef APSTUDIO_INVOKED
GUIDELINES DESIGNINFO DISCARDABLE
BEGIN
    IDD_DIALOG1, DIALOG
    BEGIN
        LEFTMARGIN, 7
        RIGHTMARGIN, 179
        TOPMARGIN, 7
        BOTTOMMARGIN, 88
    END
END
#endif // APSTUDIO_INVOKED

#endif // English (U.S.) resources
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//
////////////////////////////////////
//
// not APSTUDIO_INVOKED

```

isapi_dll/src/tpcc.cpp

```

/*      FILE:          TPCC.C          Microsoft TPC-C
 *
 *      Kit Ver. 4.20.000              Copyright
 *
 *      Microsoft, 1999                All Rights Reserved
 *
 *                                     Version
 *
 *      4.10.000 audited by Richard Gimarc, Performance
 *      Metrics, 3/17/99

```

```

 *
 *      PURPOSE: Main module for TPCC.DLL which is
 *      an ISAPI service dll.
 *      Contact: Charles Levine
 *      (clevine@microsoft.com)
 *
 *      Change history:
 *      4.20.000 - reworked error handling;
 *      added options for COM and Encina txn monitors
 *
#include <windows.h>
#include <process.h>
#include <tchar.h>
#include <stdio.h>
#include <stdarg.h>
#include <malloc.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys\timeb.h>
#include <io.h>
#include <assert.h>

#include <sqltypes.h>

#ifdef ICECAP
#include <icapexp.h>
#endif

#include "..\..\common\src\trans.h"
//tpckit transaction header contains
//definitions of structures specific to TPC-C
#include "..\..\common\src\error.h"
#include "..\..\common\src\txn_base.h"
#include "..\..\common\src\ReadRegistry.h"

#include "..\..\common\txnlog\include\rtetime.h"
#include "..\..\common\txnlog\include\spinlock.h"
#include "..\..\common\txnlog\include\txnlog.h"

// Database layer includes
#include "..\..\db_dblib_dll\src\tpcc_dblib.h"
// DBLIB implementation of TPC-C txns
#include "..\..\db_odbc_dll\src\tpcc_odbc.h"
// ODBC implementation of TPC-C txns

// Txn monitor layer includes
#include "..\..\tm_com_dll\src\tpcc_com.h"
// COM Services implementation on
// TPC-C txns
#include "..\..\tm_tuxedo_dll\src\tpcc_tux.h"
// interface to Tuxedo libraries
#include "..\..\tm_encina_dll\src\tpcc_enc.h"
// interface to Encina libraries

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

#define LEN_ERR_STRING 256

// defines for Make<Txn>Form calls to distinguish input
// and output flavors
#define OUTPUT_FORM 0
#define INPUT_FORM 1

char
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

```

```

//Terminal client id structure
TERM Term = { 0, 0, 0, NULL };

// The WEBCLIENT_VERSION string specifies the version
// level of this web client interface.
// The RTE must be synchronized with the interface
// level on login, otherwise the login
// will fail. This is a sanity check to catch problems
// resulting from mismatched versions
// of the RTE and web client.
#define WEBCLIENT_VERSION "410"

static CRITICAL_SECTION
TermCriticalSection;

static HINSTANCE hLibInstanceTm = NULL;
static HINSTANCE hLibInstanceDb = NULL;

TYPE_CTPCC_DBLIB *pCTPCC_DBLIB_new;
TYPE_CTPCC_ODBC *pCTPCC_ODBC_new;
TYPE_CTPCC_TUXEDO *pCTPCC_TUXEDO_new;
TYPE_CTPCC_ENCINA *pCTPCC_ENCINA_new;
TYPE_CTPCC_ENCINA *pCTPCC_ENCINA_post_init;
TYPE_CTPCC_COM *pCTPCC_COM_new;

// For deferred Delivery txns:
CTxnLog
*txnDelilog = NULL;
//used to log delivery transaction
information

HANDLE hWorkerSemaphore = INVALID_HANDLE_VALUE;
HANDLE hDoneEvent =
INVALID_HANDLE_VALUE;
HANDLE *pDelHandles = NULL;

// configuration settings from registry
TPCCREGISTRYDATA Reg;

DWORD dwNumDeliveryThreads = 4;
CRITICAL_SECTION DelBuffCriticalSection;
//critical section for delivery
transactions cache *pDelBuff
DELIVERY_TRANSACTION
= NULL;
DWORD dwDelBuffSize = 100;
// size of circular buffer for delivery txns
DWORD dwDelBuffFreeCount;
// number of buffers free
DWORD dwDelBuffBusyIndex = 0;
// index position of entry waiting to be delivered
DWORD dwDelBuffFreeIndex = 0;
// index position of unused entry

#include "..\..\common\src\ReadRegistry.cpp"

/* FUNCTION: DllMain
 * PURPOSE: This function is the entry point
 * for the DLL. This implementation is based on the

```

```

* fact that
DLL_PROCESS_ATTACH is only called from the inet service
once.
* ARGUMENTS: HANDLE hModule
              module handle
* ul_reason_for_call reason for call
* lpReserved reserved for future use
* RETURNS: BOOL FALSE
           errors occurred in
initialization
* TRUE DLL
successfully initialized
*/
BOOL WINAPI DllMain(HANDLE hModule, DWORD
ul_reason_for_call, LPVOID lpReserved)
{
    DWORD i;
    char szEvent[LEN_ERR_STRING] = "\\0";
    char szLogFile[128];
    char szDllName[128];

    // debugging...
    // DebugBreak();

    try
    {
        switch( ul_reason_for_call )
        {
            case DLL_PROCESS_ATTACH:
                {
                    DWORD
                    dwSize = MAX_COMPUTERNAME_LENGTH+1;
                    GetComputerName(szMyComputerName, &dwSize);
                    szMyComputerName[dwSize] = 0; }

                DisableThreadLibraryCalls((HMODULE)hModule);
                InitializeCriticalSection(&TermCriticalSection);

                if (
                ReadTPCCRegistrySettings( &Reg ) ) throw
                new CWEBCLNT_ERR( ERR_MISSING_REGISTRY_ENTRIES );

                dwDelBuffSize =
                min( Reg.dwMaxPendingDeliveries, 10000 ); // min with
                10000 as a sanity constraint

                dwNumDeliveryThreads = min(
                Reg.dwNumberOfDeliveryThreads, 100 ); // min with 100
                as a sanity constraint

                TermInit();
                // load DLL for
                if (Reg.eTxnMon
                == TUXEDO)
                {

```

```

                strcpy( szDllName, Reg.szPath );
                strcat( szDllName, "tpcc_tuxedo.dll");
                hLibInstanceTm = LoadLibrary( szDllName );
                if
                (hLibInstanceTm == NULL)
                    throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
                szDllName, GetLastError() );

                //
                get function pointer to wrapper for class constructor
                pCTPCC_TUXEDO_new = (TYPE_CTPCC_TUXEDO*)
                GetProcAddress(hLibInstanceTm, "CTPCC_TUXEDO_new");
                if
                (pCTPCC_TUXEDO_new == NULL)
                    throw new CWEBCLNT_ERR(
                ERR_GETPROCADDR_FAILED, szDllName, GetLastError() );
                else if
                (Reg.eTxnMon == ENCINA)
                {
                    strcpy( szDllName, Reg.szPath );
                    strcat( szDllName, "tpcc_encina.dll");
                    hLibInstanceTm = LoadLibrary( szDllName );
                    if
                    (hLibInstanceTm == NULL)
                        throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
                szDllName, GetLastError() );
                    //
                    get function pointer to wrapper for class constructor
                    pCTPCC_ENCINA_new = (TYPE_CTPCC_ENCINA*)
                    GetProcAddress(hLibInstanceTm, "CTPCC_ENCINA_new");
                    pCTPCC_ENCINA_post_init =
                    (TYPE_CTPCC_ENCINA*)
                    GetProcAddress(hLibInstanceTm, "CTPCC_ENCINA_post_init");
                    if
                    (pCTPCC_ENCINA_new == NULL)
                        throw new CWEBCLNT_ERR(
                ERR_GETPROCADDR_FAILED, szDllName, GetLastError() );
                    else if
                    (Reg.eTxnMon == COM)
                    {
                        strcpy( szDllName, Reg.szPath );
                        strcat( szDllName, "tpcc_com.dll");
                        hLibInstanceTm = LoadLibrary( szDllName );
                        if
                        (hLibInstanceTm == NULL)
                            throw new CWEBCLNT_ERR( ERR_LOADDLL_FAILED,
                szDllName, GetLastError() );
                    //
                    get function pointer to wrapper for class constructor

```

```

                pCTPCC_COM_new = (TYPE_CTPCC_COM*)
                GetProcAddress(hLibInstanceTm, "CTPCC_COM_new");
                if
                (pCTPCC_COM_new == NULL)
                    throw new CWEBCLNT_ERR(
                ERR_GETPROCADDR_FAILED, szDllName, GetLastError() );

                // load DLL for
                database connection
                if
                ((Reg.eTxnMon == None) || (dwNumDeliveryThreads > 0))
                {
                    if
                    (Reg.eDB_Protocol == DBLIB)
                    {
                        strcpy( szDllName, Reg.szPath );
                        strcat( szDllName, "tpcc_dblib.dll");
                        hLibInstanceDb = LoadLibrary( szDllName );
                        if (hLibInstanceDb == NULL)
                            throw new CWEBCLNT_ERR(
                ERR_LOADDLL_FAILED, szDllName, GetLastError() );

                        // get function pointer to wrapper for class
                        constructor
                        pCTPCC_DBLIB_new = (TYPE_CTPCC_DBLIB*)
                        GetProcAddress(hLibInstanceDb, "CTPCC_DBLIB_new");
                        if (pCTPCC_DBLIB_new == NULL)
                            throw new CWEBCLNT_ERR(
                ERR_GETPROCADDR_FAILED, szDllName, GetLastError() );
                        else
                        {
                            strcpy( szDllName, Reg.szPath );
                            strcat( szDllName, "tpcc_odbc.dll");
                            hLibInstanceDb = LoadLibrary( szDllName );
                            if (hLibInstanceDb == NULL)
                                throw new CWEBCLNT_ERR(
                ERR_LOADDLL_FAILED, szDllName, GetLastError() );

                            // get function pointer to wrapper for class
                            constructor
                            pCTPCC_ODBC_new = (TYPE_CTPCC_ODBC*)
                            GetProcAddress(hLibInstanceDb, "CTPCC_ODBC_new");
                            if (pCTPCC_ODBC_new == NULL)
                                throw new CWEBCLNT_ERR(
                ERR_GETPROCADDR_FAILED, szDllName, GetLastError() );
                        }
                    }
                }
            }
        }
    }
}

```

```

(dwNumDeliveryThreads)
    if
    {
        //
for deferred delivery txns:
    hDoneEvent = CreateEvent( NULL, TRUE /*
manual reset */, FALSE /* initially not signalled */,
NULL );
    InitializeCriticalSection(&DelBuffCriticalSec
tion);
    hWorkerSemaphore = CreateSemaphore( NULL, 0,
dwDelBuffSize, NULL );
    dwDelBuffFreeCount = dwDelBuffSize;

    InitJulianTime(NULL);

create unique log file name based on delilog-yyymmdd-
hhmm.log
    SYSTEMTIME Time;
    GetLocalTime( &Time );
    wsprintf( szLogFile, "%sdelivery-
%2.2d%2.2d-%2.2d-%2.2d.log",
        Reg.szPath, Time.wYear % 100,
Time.wMonth, Time.wDay, Time.wHour, Time.wMinute );
    txnDelilog = new CTxnLog(SzLogFile,
TXN_LOG_WRITE);

    //write event into txn log for START
    txnDelilog-
>writeCtrlRectoLog(TXN_EVENT_START, szMyComputerName,
sizeof(szMyComputerName));

    //
allocate structures for delivery buffers and thread
mgmt
    pDelihandles = new
HANDLE[dwNumDeliveryThreads];
    pDelBuff = new
DELIVERY_TRANSACTION[dwDelBuffSize];
    //
launch DeliveryworkerThread to perform actual delivery
txns
    for(i=0; i<dwNumDeliveryThreads; i++)
    {
        pDelihandles[i] = (HANDLE) _beginthread(
DeliveryworkerThread, 0, NULL );
        if (pDelihandles[i] == INVALID_HANDLE_VALUE)
            throw new CWEBCLNT_ERR(
ERR_DELIVERY_THREAD_FAILED );
    }
    break;

```

```

case DLL_PROCESS_DETACH:
    if
    {
        if
        {
            //write event into txn log for STOP
            txnDelilog->writeCtrlRectoLog(TXN_EVENT_STOP,
szMyComputerName, sizeof(szMyComputerName));
            // This will do a clean shutdown of the
delivery log file
            CTxnLog *txnDelilogLocal = txnDelilog;
            txnDelilog= NULL;
            delete txnDelilogLocal;
        }

        delete [] pDelihandles;
        delete [] pDelBuff;

        CloseHandle( hWorkerSemaphore );
        CloseHandle( hDoneEvent );
        DeleteCriticalSection(&DelBuffCriticalSection
);
    }

    DeleteCriticalSection(&TermCriticalSection);

    if
    {
        if
        {
            FreeLibrary( hLibInstanceTm );
            hLibInstanceTm
= NULL;
        }
        if
        {
            FreeLibrary( hLibInstanceDb );
            hLibInstanceDb
= NULL;
            Sleep(500);
            break;
        }
        default: /* nothing */;
    }
}
catch (CBaseErr *e)
{
    writeMessageToEventLog( e-
>ErrorText() );
    delete e;
    TerminateExtension(0);
    return FALSE;
}
catch (...)

```

```

{
    writeMessageToEventLog(TEXT("Unhandled
exception. DLL could not load."));
    TerminateExtension(0);
    return FALSE;
}
return TRUE;

/* FUNCTION: GetExtensionVersion
* PURPOSE: This function is called by the inet
service when the DLL is first loaded.
* ARGUMENTS: HSE_VERSION_INFO *pver
passed in structure in which to place
expected version number.
* RETURNS: TRUE inet service
expected return value.
*/
BOOL WINAPI GetExtensionVersion(HSE_VERSION_INFO *pver)
{
    pver->dwExtensionVersion =
MAKELONG(HSE_VERSION_MINOR, HSE_VERSION_MAJOR);
    lstrcpy(pver->pszExtensionDesc, "TPC-C
Server.", HSE_MAX_EXT_DLL_NAME_LEN);

    // TODO: why do we need this here instead of
in the DLL attach?
    if (Reg.eTxnMon == ENCINA)
        pCTPCC_ENCINA_post_init();

    return TRUE;
}

/* FUNCTION: TerminateExtension
* PURPOSE: This function is called by the inet
service when the DLL is about to be unloaded.
Release all resources in
anticipation of being unloaded.
* RETURNS: TRUE inet service
expected return value.
*/
BOOL WINAPI TerminateExtension( DWORD dwFlags )
{
    if (pDelihandles)
    {
        SetEvent( hDoneEvent );
        for(DWORD i=0;
i<dwNumDeliveryThreads; i++)
            WaitForSingleObject(
pDelihandles[i], INFINITE );
    }

    TermDeleteAll();
    return TRUE;
}

/* FUNCTION: HttpExtensionProc
* PURPOSE: This function is the main entry
point for the TPCC DLL. The internet service

```

```

*           calls this function
passing in the http string.
* ARGUMENTS:  EXTENSION_CONTROL_BLOCK *pECB
              structure pointer to passed in internet
*
              service information.
* RETURNS:    DWORD
              HSE_STATUS_SUCCESS
              connection can be dropped if error
              HSE_STATUS_SUCCESS_AND_KEEP_CONN keep
connect valid comment sent
* COMMENTS:   None
*/
DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK
*pECB)
{
    int iCmd, FormId,
    char szBuffer[4096];

    int lpbSize;
    static char szHeader[] = "200 Ok";
    DWORD dwSize = 6;
    // initial value is strlen(szHeader)
    char szHeader1[4096];

#ifdef ICECAP
    StartCAP();
#endif

    try
    {
        //process http query
        ProcessQueryString(pECB, &iCmd,
        &FormId, &TermId, &iSyncId);

        if (TermId != 0)
        {
            if (TermId < 0 || TermId
            >= Term.iNumEntries ||
            Term.pClientData[TermId].iNextFree != -1)
            {
                // debugging...
                char szTmp[128];
                wsprintf(
                szTmp, "Invalid term ID; TermId = %d", TermId);
                WriteMessageToEventLog( szTmp );
                throw new
                CWEBCLNT_ERR( ERR_INVALID_TERMID );
            }
            //must have a valid
            syncid here since termid is valid
            if (iSyncId !=
            Term.pClientData[TermId].iSyncId)
            throw new
            CWEBCLNT_ERR( ERR_INVALID_SYNC_CONNECTION );
            //set use time

```

```

        Term.pClientData[TermId].iTickCount =
        GetTickCount();
    }

    switch(iCmd)
    {
    case 0:
        WelcomeForm(pECB,
        szBuffer);
        break;
    case 1:
        switch( FormId )
        {
        case
        WELCOME_FORM:
        case
        MAIN_MENU_FORM:
        break;
        case
        NEW_ORDER_FORM:
        ProcessNewOrderForm(pECB, TermId, szBuffer);
        break;
        case
        PAYMENT_FORM:
        ProcessPaymentForm(pECB, TermId, szBuffer);
        break;
        case
        DELIVERY_FORM:
        ProcessDeliveryForm(pECB, TermId, szBuffer);
        break;
        case
        ORDER_STATUS_FORM:
        ProcessOrderStatusForm(pECB, TermId,
        szBuffer);
        break;
        case
        STOCK_LEVEL_FORM:
        ProcessStockLevelForm(pECB, TermId,
        szBuffer);
        break;
        }
    case 2:
        // new-order selected
        from menu; display new-order input form
        MakeNewOrderForm(TermId,
        NULL, INPUT_FORM, szBuffer);
        break;
    case 3:
        // payment selected from
        menu; display payment input form
        MakePaymentForm(TermId,
        NULL, INPUT_FORM, szBuffer);
        break;
    case 4:
        // delivery selected from
        menu; display delivery input form
        MakeDeliveryForm(TermId,
        NULL, INPUT_FORM, szBuffer);

```

```

        case 5:
        break;
        // order-status selected
        from menu; display order-status input form
        MakeOrderStatusForm(TermId, NULL, INPUT_FORM,
        szBuffer);
        break;
    case 6:
        // stock-level selected
        from menu; display stock-level input form
        MakeStockLevelForm(TermId, NULL, INPUT_FORM,
        szBuffer);
        break;
    case 7:
        // ExitCmd
        TermDelete(TermId);
        WelcomeForm(pECB,
        szBuffer);
        break;
    case 8:
        SubmitCmd(pECB,
        szBuffer);
        break;
    case 9:
        // menu
        MakeMainMenuForm(TermId,
        Term.pClientData[TermId].iSyncId, szBuffer);
        break;
    case 10:
        // CMD=Clear
        // resets all
        connections; should only be used when no other
        connections are active
        TermDeleteAll();
        TermInit();
        WelcomeForm(pECB,
        szBuffer);
        break;
    case 11:
        // CMD=Stats
        StatsCmd(pECB, szBuffer);
        break;
    }
    catch (CBaseErr *e)
    {
        ErrorForm( pECB, e->ErrorType(), e-
        >ErrorNum(), TermId, iSyncId, e->ErrorText(), szBuffer
        );
        delete e;
    }
    catch (...)
    {
        ErrorForm( pECB, ERR_TYPE_WEBDLL,
        0, TermId, iSyncId, "Error: unhandled exception in web
        Client.", szBuffer );
    }
#ifdef ICECAP
    StopCAP();
#endif

    lpbSize = strlen(szBuffer);
    wsprintf(szHeader1, "Content-Type:
    text/html\r\n"
    "Content-Length: %d\r\n"
    "Connection: Keep-
    Alive\r\n\r\n", lpbSize);
    strcat( szHeader1, szBuffer );

```

```

        (*pECB->ServerSupportFunction)(pECB->ConnID,
HSE_REQ_SEND_RESPONSE_HEADER, szHeader, (LPDWORD)
&dwSize, (LPDWORD)szHeader1);

        //finish up and keep connection
pECB->dwHttpStatusCode = 200;
return HSE_STATUS_SUCCESS_AND_KEEP_CONN;
}

```

```

void writeMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];

    // Use event logging to log the error.
    //
    hEventSource = RegisterEventSource(NULL,
TEXT("TPCC.DLL"));

    _stprintf(szMsg, TEXT("Error in TPCC.DLL: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event
source
        EVENTLOG_ERROR_TYPE, // event type
        0, // event Category
        0, // event ID
        NULL, // current user's SID
        2, // strings in
lpszStrings
        0, // no bytes of raw
data
        (LPCTSTR *)lpszStrings, // array of
error strings
        NULL); // no raw data

        (VOID) DeregisterEventSource(hEventSource);
    }
}

```

```

/* FUNCTION: DeliveryworkerThread
*
* PURPOSE: This function processes deferred
delivery txns. There are typically several
* threads running this
routine. The number of threads is determined by an
entry
* read from the registry.
The thread waits for work by waiting on semaphore.
* When a delivery txn is
posted, the semaphore is released. After processing
* the delivery txn,
information is logged to record the txn status and
execution
* time.
*/

```

```

/*static*/ void DeliveryworkerThread(void *ptr)
{
    CTPCC_BASE *pTxn = NULL;

    DELIVERY_TRANSACTION
delivery;
PDELIVERY_DATA
pDeliveryData;

```

```

TXN_RECORD_TPCC_DELIV_DEF txnDelirec;

DWORD
index;
HANDLE
handles[2];

SYSTEMTIME trans_end;
//delivery transaction finished
time
SYSTEMTIME trans_start;
//delivery transaction start time

assert(txnDelilog != NULL);

try
{
    if (Reg.eDB_Protocol == ODBC)
        pTxn = pCTPCC_ODBC_new(
Reg.szDbServer, Reg.szDbuser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName, Reg.szSPPrefix );
    else if (Reg.eDB_Protocol == DBLIB)
        pTxn = pCTPCC_DBLIB_new(
Reg.szDbServer, Reg.szDbuser, Reg.szDbPassword,
szMyComputerName, Reg.szDbName );
    pDeliveryData = pTxn-
>BuffAddr_Delivery();
}
catch (CBaseErr *e)
{
    char szTmp[1024];
    wsprintf( szTmp, "Error in Delivery
Txn thread. Could not connect to database. "
Server=%s, User=%s, Password=%s, Database=%s",
e->ErrorText(),
Reg.szDbServer, Reg.szDbuser, Reg.szDbPassword,
Reg.szDbName );
    writeMessageToEventLog( szTmp );
    delete e;
    goto ErrorExit;
}
catch (...)
{
    writeMessageToEventLog(TEXT("Unhandled
exception caught in DeliveryworkerThread."));
    goto ErrorExit;
}

while (TRUE)
{
    try
    {
        //while delivery thread
running, i.e. user has not requested termination
while (TRUE)
        {
            // need to wait
for multiple objects: program exit or worker semaphore;
handles[0] =
hDoneEvent;
handles[1] =
hWorkerSemaphore;
index =
waitForMultipleObjects( 2, &handles[0], FALSE, INFINITE
);
if (index ==
WAIT_OBJECT_0)
                goto
ErrorExit;

```

```

        ZeroMemory(&txnDelirec, sizeof(txnDelirec));
        txnDelirec.TxnType =
TXN_REC_TYPE_TPCC_DELIV_DEF;

        // make a local
copy of current entry from delivery buffer and
increment buffer index
        EnterCriticalSection(&DelBuffCriticalSection)
;
        delivery =
*(pDelBuff+dwDelBuffBusyIndex);
        dwDelBuffFreeCount++;
        dwDelBuffBusyIndex++;
        if
(dwDelBuffBusyIndex == dwDelBuffSize) // wrap-around
if at end of buffer
            dwDelBuffBusyIndex = 0;

        LeaveCriticalSection(&DelBuffCriticalSection)
;

        pDeliveryData-
>w_id = delivery.w_id;
        pDeliveryData-
>o_carrier_id = delivery.o_carrier_id;
        txnDelirec.w_id
= pDeliveryData->w_id;
        txnDelirec.o_carrier_id = pDeliveryData-
>o_carrier_id;
        txnDelirec.TxnStartT0 =
Get64BitTime(&delivery.queueue);

        GetLocalTime(
        &trans_start );
        pTxn-
>Delivery();
        GetLocalTime(
        &trans_end );

        //log txn
        txnDelirec.TxnStatus = ERR_SUCCESS;
        for (int i=0;
i<10; i++)
            txnDelirec.o_id[i] = pDeliveryData->o_id[i];

        txnDelirec.DeltaT4 =
(int)(Get64BitTime(&trans_end) -
txnDelirec.TxnStartT0);

        txnDelirec.DeltaTxnExec =
(int)(Get64BitTime(&trans_end) -
Get64BitTime(&trans_start));

        if (txnDelilog
!= NULL)
            if (txnDelilog
                txnDelilog->WriteToLog(&txnDelirec);
            }
        }
    }
    catch (CBaseErr *e)

```

```

        {
            char szTmp[1024];
            wsprintf( szTmp, "Error
in Delivery Txn thread. %s",
e->ErrorText() );
            WriteMessageToEventLog(
szTmp );

            // log the error txn
            txnDelRec.TxnStatus = e-
>ErrorType();
            if (txnDelRec != NULL)
                txnDelRec->
                delete e;
        }
        catch (...)
        {
            // unhandled exception;
            shouldn't happen; not much we can do...
            WriteMessageToEventLog(TEXT("Unhandled
exception caught in DeliveryWorkerThread.
"));
        }
    }
    ErrorExit:
        delete pTxn;
        _endthread();
    }

/* FUNCTION: PostDeliveryInfo
* PURPOSE: This function enters the delivery
txn into the deferred delivery buffer.
* RETURNS: BOOL FALSE
            delivery information posted successfully
            TRUE error cannot post delivery info
*/
BOOL PostDeliveryInfo(long w_id, short o_carrier_id)
{
    BOOL bError;
    EnterCriticalSection(&DelBuffCriticalSection)
;
    if (dwDelBuffFreeCount > 0)
    {
        bError = FALSE;
        (pDelBuff+dwDelBuffFreeIndex)->w_id
        = w_id;
        (pDelBuff+dwDelBuffFreeIndex)-
        = o_carrier_id;
        GetLocalTime(&(pDelBuff+dwDelBuffFreeIndex)-
        >queue);

        dwDelBuffFreeCount--;
        dwDelBuffFreeIndex++;
        if (dwDelBuffFreeIndex ==
dwDelBuffSize)
            dwDelBuffFreeIndex = 0;
        // wrap-around if at end of buffer
    }
    else
        // No free buffers. Return an
error, which indicates that the delivery buffer is
full.

```

```

// Most likely, the number of
delivery worker threads needs to be increased to keep
up
// with the txn rate.
bError = TRUE;
LeaveCriticalSection(&DelBuffCriticalSection)
;
    if (!bError)
        // increment worker semaphore to
wake up a worker thread
        ReleaseSemaphore( hWorkerSemaphore,
1, NULL );
    return bError;
}

/* FUNCTION: ProcessQueryString
* PURPOSE: This function extracts the relevent
information out of the http command passed in from
the browser.
* COMMENTS: If this is the initial connection
i.e. client is at welcome screen then
there will not
be a terminal id or current form id. If this is the
case
then the
pTermid and pFormid return values are undefined.
*/

void ProcessQueryString(EXTENSION_CONTROL_BLOCK *pECB,
int *pCmd, int *pFormId, int *pTermId, int *pSyncId)
{
    char *ptr = pECB->lpszQueryString;
    char szBuffer[25];
    int i;

    //allowable client command strings i.e.
    static char *szCmds[] =
    {
        "Process", "..NewOrder..",
        "..Payment..", "..Delivery..", "..Order-Status..",
        "..Stock-Level..", "..Exit..", "Submit", "Menu",
        "Clear", "Stats", ""
    };
    *pCmd = 0; // default is
the login screen
    *pTermId = 0;

    // if no params (i.e., empty query string),
then return login screen
    if (strlen(pECB->lpszQueryString) == 0)
        return;

    // parse FORMID, TERMID, and SYNCID
    *pFormId = GetIntKeyValue(&ptr, "FORMID",
NO_ERR, NO_ERR);
    *pTermId = GetIntKeyValue(&ptr, "TERMID",
NO_ERR, NO_ERR);
    *pSyncId = GetIntKeyValue(&ptr, "SYNCID",
NO_ERR, NO_ERR);

    // parse CMD
    GetKeyValue(&ptr, "CMD", szBuffer,
sizeof(szBuffer), ERR_COMMAND_UNDEFINED);

    // see which command it matches

```

```

for(i=0; ; i++)
{
    if (szCmds[i][0] == 0)
        // no more; no match;
return error
        throw new CWEBCLNT_ERR(
ERR_COMMAND_UNDEFINED );
    if ( !strcmp(szCmds[i], szBuffer) )
    {
        *pCmd = i+1;
        break;
    }
}

/* FUNCTION: void welcomeForm
*/

void welcomeForm(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer)
{
    char szTmp[1024];

    //welcome to tpc-c html form buffer, this is
first form client sees.
    strcpy( szBuffer, "<HTML><HEAD><TITLE>TPC-C
Web Client</TITLE></HEAD><BODY>"

        "<B><BIG>Microsoft TPC-C Web Client (ver
4.20)</BIG></B> <BR> <BR>"

        "<font face=\\"Courier New\\"><PRE>"
        "Compiled: \"_DATE_\", \"_TIME_\" <BR>"
        "Source: \"_FILE_\" (\"_TIMESTAMP_\") <BR>"
        "</PRE></font>"

        "<FORM ACTION=\\"tpcc.dll\\" METHOD=\\"GET\\">"
        "<INPUT TYPE=\\"hidden\\" NAME=\\"STATUSID\\"
VALUE=\\"0\\">"
        "<INPUT TYPE=\\"hidden\\" NAME=\\"ERROR\\"
VALUE=\\"0\\">"
        "<INPUT TYPE=\\"hidden\\" NAME=\\"FORMID\\"
VALUE=\\"1\\">"
        "<INPUT TYPE=\\"hidden\\" NAME=\\"TERMID\\"
VALUE=\\"0\\">"
        "<INPUT TYPE=\\"hidden\\" NAME=\\"SYNCID\\"
VALUE=\\"0\\">"
        "<INPUT TYPE=\\"hidden\\" NAME=\\"VERSION\\"
VALUE=\\"\" WEBCIENT_VERSION \"\");
        sprintf( szTmp, "configuration Settings:
<BR><font face=\\"Courier New\\" color=\\"blue\\"><PRE>"
        "Txn
Monitor = <B>%s</B><BR>"
        "Database protocol = <B>%s</B><BR>"
        "Connections = <B>%d</B><BR>"
        "Max
Delivery Threads = <B>%d</B><BR>"

```

```

Pending Deliveries = <B>%d</B><BR>"
                                                                    "Max
szTxnMonNames[Reg.eTxnMon],
szDBNames[Reg.eDB_Protocol],
                                                                    Reg.dwMaxConnections,
dwNumDeliveryThreads, dwDelBuffSize );
                                                                    strcat( szBuffer, szTmp);
                                                                    if (Reg.eTxnMon == COM)
                                                                    {
                                                                    printf( szTmp, "COM Single
Pool
                                                                    = <B>%s</B><BR>",
                                                                    Reg.bCOM_SinglePool ?
"YES" : "NO" );
                                                                    strcat( szBuffer, szTmp);
                                                                    }
                                                                    strcat( szBuffer, "</PRE></font>");
                                                                    if (Reg.eTxnMon == None)
                                                                    // connection options may be
specified when not using a txn monitor
                                                                    printf( szTmp, "Please enter
your database options for this connection:<BR>"
                                                                    color="blue"><PRE>"
                                                                    "DB Server = <INPUT NAME=\"db_server\"
SIZE=20 VALUE=\"%s\"><BR>"
                                                                    "DB User ID = <INPUT NAME=\"db_user\"
SIZE=20 VALUE=\"%s\"><BR>"
                                                                    "DB Password = <INPUT NAME=\"db_passwd\"
SIZE=20 VALUE=\"%s\"><BR>"
                                                                    "DB Name = <INPUT NAME=\"db_name\"
SIZE=20 VALUE=\"%s\"><BR>"
                                                                    "</PRE></font>"
                                                                    Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
Reg.szDbName );
                                                                    else
                                                                    // if using a txn monitor,
connection options are determined from registry; can't
                                                                    // set per user. show options fyi
                                                                    printf( szTmp, "Database
options which will be used by the transaction
monitor:<BR>"
                                                                    color="blue"><PRE>"
                                                                    "DB Server = <B>%s</B><BR>"
                                                                    "DB User ID = <B>%s</B><BR>"
                                                                    "DB Password = <B>%s</B><BR>"
                                                                    "DB Name = <B>%s</B><BR>"
                                                                    "</PRE></font>"
                                                                    Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
Reg.szDbName );
                                                                    strcat( szBuffer, szTmp);
                                                                    printf( szTmp, "Please enter your
warehouse and District for this session:<BR>"

```

```

                                                                    "<font face=\"Courier New\"
color=blue\"><PRE>" );
                                                                    strcat( szBuffer, szTmp);
                                                                    strcat( szBuffer, "warehouse ID = <INPUT
NAME=\"w_id\" SIZE=6><BR>"
                                                                    "District ID = <INPUT NAME=\"d_id\"
SIZE=2><BR>"
                                                                    "</PRE></font><HR>"
                                                                    "<INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\"Submit\">"
                                                                    "</FORM></BODY></HTML>");
                                                                    }
                                                                    /* FUNCTION: SubmitCmd
                                                                    *
                                                                    * PURPOSE: This function allocated a new
terminal id in the Term structure array.
                                                                    */
                                                                    void SubmitCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer)
                                                                    {
                                                                    int
                                                                    char *ptr = pECB->pszQueryString;
                                                                    char
                                                                    szVersion[32] = { 0 };
                                                                    char
                                                                    szServer[32] = { 0 };
                                                                    char
                                                                    szUser[32] =
                                                                    "sa";
                                                                    char
                                                                    szPassword[32] = { 0 };
                                                                    char
                                                                    szDatabase[32] = "tpcc";
                                                                    // validate version field; the version field
ensures that the RTE is synchronized with the web
client
                                                                    GetKeyValue(&ptr, "VERSION", szVersion,
sizeof(szVersion), ERR_VERSION_MISMATCH);
                                                                    if ( strcmp( szVersion, WEBCLIENT_VERSION ) )
                                                                    throw new CWEBCLNT_ERR(
ERR_VERSION_MISMATCH );
                                                                    if (Reg.eTxnMon == None)
                                                                    {
                                                                    // parse Server name
                                                                    GetKeyValue(&ptr, "db_server",
szServer, sizeof(szServer), ERR_NO_SERVER_SPECIFIED);
                                                                    // parse User name
                                                                    GetKeyValue(&ptr, "db_user",
szUser, sizeof(szUser), NO_ERR);
                                                                    // parse Password
                                                                    GetKeyValue(&ptr, "db_passwd",
szPassword, sizeof(szPassword), NO_ERR);
                                                                    // parse Database name
                                                                    GetKeyValue(&ptr, "db_name",
szDatabase, sizeof(szDatabase), NO_ERR);
                                                                    }
                                                                    // parse warehouse ID
                                                                    int w_id = GetIntKeyValue(&ptr, "w_id",
ERR_HTML_ILL_FORMED, ERR_W_ID_INVALID);
                                                                    if ( w_id < 1 )
                                                                    throw new CWEBCLNT_ERR(
ERR_W_ID_INVALID );
                                                                    // parse district ID

```

```

                                                                    int d_id = GetIntKeyValue(&ptr, "d_id",
ERR_HTML_ILL_FORMED, ERR_D_ID_INVALID);
                                                                    if ( d_id < 1 || d_id > 10 )
                                                                    throw new CWEBCLNT_ERR(
ERR_D_ID_INVALID );
                                                                    iNewTerm = TermAdd();
                                                                    Term.pClientData[iNewTerm].w_id = w_id;
                                                                    Term.pClientData[iNewTerm].d_id = d_id;
                                                                    try
                                                                    {
                                                                    if (Reg.eTxnMon == TUXEDO)
                                                                    Term.pClientData[iNewTerm].pTxn =
pCTPCC_TUXEDO_new();
                                                                    else if (Reg.eTxnMon == ENCINA)
                                                                    Term.pClientData[iNewTerm].pTxn =
pCTPCC_ENCINA_new();
                                                                    else if (Reg.eTxnMon == COM)
                                                                    Term.pClientData[iNewTerm].pTxn =
pCTPCC_COM_new( Reg.bCOM_SinglePool );
                                                                    else if (Reg.eDB_Protocol == ODBC)
                                                                    Term.pClientData[iNewTerm].pTxn =
pCTPCC_ODBC_new( szServer, szUser, szPassword,
szMyComputerName, szDatabase, Reg.szSPPrefix );
                                                                    else if (Reg.eDB_Protocol == DBLIB)
                                                                    Term.pClientData[iNewTerm].pTxn =
pCTPCC_DBLIB_new( szServer, szUser, szPassword,
szMyComputerName, szDatabase );
                                                                    }
                                                                    catch (...)
                                                                    {
                                                                    TermDelete(iNewTerm);
                                                                    throw; // pass
                                                                    }
                                                                    exception upward
                                                                    }
                                                                    MakeMainMenuForm(iNewTerm,
Term.pClientData[iNewTerm].iSyncId, szBuffer);
                                                                    }
                                                                    /* FUNCTION: StatsCmd
                                                                    *
                                                                    * PURPOSE: This function returns to the
browser the total number of active terminal ids.
                                                                    * This routine is for
development/debugging purposes.
                                                                    */
                                                                    void StatsCmd(EXTENSION_CONTROL_BLOCK *pECB, char
*szBuffer)
                                                                    {
                                                                    int i;
                                                                    int iTotals;
                                                                    EnterCriticalSection(&TermCriticalSection);
                                                                    iTotals = 0;
                                                                    for(i=0; i<Term.iNumEntries; i++)
                                                                    {
                                                                    if (Term.pClientData[i].iNextFree
== -1)
                                                                    iTotals++;
                                                                    }

```



```

LeaveCriticalSection(&TermCriticalSection);
wsprintf( szBuffer,
"Web Client Stats</TITLE></HEAD>"
"Active Connections: %d </BIG></B><BR></BODY></HTML>"
);
char *CWEBCNT_ERR::ErrorText()
{
static SERRORMSG errorMsgs[] =
{
{ ERR_COMMAND_UNDEFINED,
"Command undefined."
},
{ ERR_D_ID_INVALID,
"Invalid District ID Must be 1 to 10."
},
{ ERR_DELIVERY_CARRIER_ID_RANGE,
"Delivery Carrier ID out of range
must be 1 - 10."
},
{ ERR_DELIVERY_CARRIER_INVALID,
"Delivery Carrier ID invalid must be numeric
1 - 10."
},
{ ERR_DELIVERY_MISSING_OCD_KEY,
"Delivery missing Carrier ID key \"OCD*\"."
},
{ ERR_DELIVERY_THREAD_FAILED,
"Could not start delivery worker
thread."
},
{ ERR_GETPROCADDR_FAILED,
"Could not map proc in DLL. GetProcAddr
error. DLL="
},
{ ERR_HTML_ILL_FORMED,
"Required key field is missing from HTML
string."
},
{ ERR_INVALID_SYNC_CONNECTION,
"Invalid Terminal Sync ID."
},
{ ERR_INVALID_TERMID,
"Invalid Terminal ID."
},
{ ERR_LOADDLL_FAILED,
"Load
of DLL failed. DLL="
},
{ ERR_MAX_CONNECTIONS_EXCEEDED,
"No connections available. Max Connections
is probably too low."
}
}
}

```

```

{
ERR_MISSING_REGISTRY_ENTRIES,
"Required registry entries are missing.
Rerun INSTALL to correct."
},
{ ERR_NEWORDER_CUSTOMER_INVALID,
"New Order customer id invalid data
type, range = 1 to 3000."
},
{ ERR_NEWORDER_CUSTOMER_KEY,
"New Order missing Customer key
\"CID*\"."
},
{ ERR_NEWORDER_DISTRICT_INVALID,
"New Order District ID Invalid
range 1 - 10."
},
{ ERR_NEWORDER_FORM_MISSING_DID,
"New Order missing District key
\"DID*\"."
},
{ ERR_NEWORDER_ITEMID_INVALID,
"New Order Item Id is wrong data type, must
be numeric."
},
{ ERR_NEWORDER_ITEMID_RANGE,
"New Order Item Id is out of range.
Range = 1 to 999999."
},
{ ERR_NEWORDER_ITEMID_WITHOUT_SUPPW,
"New Order Item Id field entered without a
corresponding Supp_w."
},
{ ERR_NEWORDER_MISSING_IID_KEY,
"New Order missing Item Id key \"IID*\"."
},
{ ERR_NEWORDER_MISSING_QTY_KEY,
"New Order Missing Qty key \"Qty##*\"."
},
{ ERR_NEWORDER_MISSING_SUPPW_KEY,
"New Order missing Supp_w key
\"SP##*\"."
},
{ ERR_NEWORDER_NOITEMS_ENTERED,
"New Order No order lines entered."
},
{ ERR_NEWORDER_QTY_INVALID,
"New Order Qty
invalid must be numeric range 1 - 99."
},
{ ERR_NEWORDER_QTY_RANGE,
"New
Order Qty is out of range. Range = 1 to 99."
},
{ ERR_NEWORDER_QTY_WITHOUT_SUPPW,
"New Order Qty field entered
without a corresponding Supp_w."
},
{ ERR_NEWORDER_SUPPW_INVALID,
"New Order Supp_w invalid data type
must be numeric."
}
}
}

```

```

{
ERR_NO_SERVER_SPECIFIED,
"No Server name
specified."
},
{ ERR_ORDERSTATUS_CID_AND_CLT,
"Order Status Only Customer ID or Last Name
may be entered, not both."
},
{ ERR_ORDERSTATUS_CID_INVALID,
"Order Status Customer ID invalid, range must
be numeric 1 - 3000."
},
{ ERR_ORDERSTATUS_CLT_RANGE,
"Order Status Customer last name
longer than 16 characters."
},
{ ERR_ORDERSTATUS_DID_INVALID,
"Order Status District invalid, value must be
numeric 1 - 10."
},
{ ERR_ORDERSTATUS_MISSING_CID_CLT,
"Order Status Either Customer ID or Last Name
must be entered."
},
{ ERR_ORDERSTATUS_MISSING_CID_KEY,
"Order Status missing Customer key \"CID*\"."
},
{ ERR_ORDERSTATUS_MISSING_CLT_KEY,
"Order Status missing Customer Last Name key
\"CLT*\"."
},
{ ERR_ORDERSTATUS_MISSING_DID_KEY,
"Order Status missing District key \"DID*\"."
},
{ ERR_PAYMENT_CDI_INVALID,
"Payment
Customer district invalid must be numeric."
},
{ ERR_PAYMENT_CID_AND_CLT,
"Payment Only
Customer ID or Last Name may be entered, not both."
},
{ ERR_PAYMENT_CUSTOMER_INVALID,
"Payment Customer data type invalid, must be
numeric."
},
{ ERR_PAYMENT_CWI_INVALID,
"Payment
Customer warehouse invalid, must be numeric."
},
{ ERR_PAYMENT_DISTRICT_INVALID,
"Payment District ID is invalid, must be 1 -
10."
},
{ ERR_PAYMENT_HAM_INVALID,
"Payment Amount
invalid data type must be numeric."
},
{ ERR_PAYMENT_HAM_RANGE,
"Payment Amount out of range, 0 - 9999.99."
},
{ ERR_PAYMENT_LAST_NAME_TO_LONG,
"Payment Customer last name longer

```

```

than 16 characters."
    },
    {
        ERR_PAYMENT_MISSING_CDI_KEY,
        "Payment missing Customer district key
        \"CDI*\"."
    },
    {
        ERR_PAYMENT_MISSING_CID_CLT,
        "Payment Either Customer ID or Last Name must
        be entered."
    },
    {
        ERR_PAYMENT_MISSING_CID_KEY,
        "Payment missing Customer key \"CID*\"."
    },
    {
        ERR_PAYMENT_MISSING_CLT_KEY,
        "Payment missing Customer Last Name key
        \"CLT*\"."
    },
    {
        ERR_PAYMENT_MISSING_CWI_KEY,
        "Payment missing Customer warehouse key
        \"CWI*\"."
    },
    {
        ERR_PAYMENT_MISSING_DID_KEY,
        "Payment missing District Key \"DID*\"."
    },
    {
        ERR_PAYMENT_MISSING_HAM_KEY,
        "Payment missing Amount key \"HAM*\"."
    },
    {
        ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
        "Stock Level; missing Threshold key \"TT*\"."
    },
    {
        ERR_STOCKLEVEL_THRESHOLD_INVALID,
        "Stock Level; Threshold value must be in the
        range = 1 - 99."
    },
    {
        ERR_STOCKLEVEL_THRESHOLD_RANGE,
        "Stock Level Threshold out of
        range, range must be 1 - 99."
    },
    {
        ERR_VERSION_MISMATCH,
        "Invalid version field. RTE and Web Client
        are probably out of sync."
    },
    {
        ERR_W_ID_INVALID,
        "Invalid warehouse ID."
    },
    {
        0,
        ""
    },
    }
};

char szTmp[256];
int i = 0;
while (TRUE)
{
    if (errorMsgs[i].szMsg[0] == 0)

```

```

        strcpy( szTmp, "Unknown
        error number." );
        break;
    }
    if (m_Error == errorMsgs[i].iError)
    {
        strcpy( szTmp,
        errorMsgs[i].szMsg );
        break;
    }
    i++;
}

if (m_szTextDetail)
    strcat( szTmp, m_szTextDetail );
if (m_SystemErr)
    sprintf( szTmp+strlen(szTmp), "
Error=%d", m_SystemErr );

m_szErrorText = new char[strlen(szTmp)+1];
strcpy( m_szErrorText, szTmp );
return m_szErrorText;
}

/* FUNCTION: GetKeyValue
* PURPOSE: This function parses a http
formatted string for specific key values.
* ARGUMENTS: char http string from client
browser *pQueryString char key
value to look for *pkey char key
* character array into which to place key's
value *pvalue char
* iMax int
* maximum length of key value array.
err WEBERROR error
value to throw
* RETURNS: nothing.
* ERROR: if (the pkey value is not found)
then if
(err == 0) if
* return (empty string)
* else
* throw CWEBCLNT_ERR(err)
* COMMENTS: http keys are formatted either
KEY=value& or KEY=value\0. This DLL formats
TPC-C input
fields in such a manner that the keys can be extracted
in the above manner.
*/

void GetKeyValue(char **pQueryString, char *pkey, char
*pvalue, int iMax, WEBERROR err)
{
    char *ptr;

```

```

    if ( !(ptr=strstr(*pQueryString, pkey)) )
        goto ErrorExit;
    ptr += strlen(pkey);
    if ( *ptr != '=' )
        goto ErrorExit;
    ptr++;
    iMax--; // one position is for terminating
null
while( *ptr && *ptr != '&' && iMax)
{
    *pvalue++ = *ptr++;
    iMax--;
}
*pvalue = 0; // terminating null
*pQueryString = ptr;
return;

ErrorExit:
if (err != NO_ERR)
    throw new CWEBCLNT_ERR( err );
*pvalue = 0; // return empty result string
}

/* FUNCTION: GetIntKeyValue
* PURPOSE: This function parses a http
formatted string for a specific key value.
* ARGUMENTS: char http string from client
browser *pQueryString char key
value to look for *pkey char key
* NoKeyErr WEBERROR error value to throw if
key not found
* NotIntErr WEBERROR error value to throw if
value not numeric
* RETURNS: integer
* ERROR: if (the pkey value is not found)
then if
(NoKeyErr != NO_ERR) if
* throw CWEBCLNT_ERR(err)
* else
* return 0
else if (non-
numeric char found) then
if
(NotIntErr != NO_ERR) then
* throw CWEBCLNT_ERR(err)
* else
* return 0
* COMMENTS: http keys are formatted either
KEY=value& or KEY=value\0. This DLL formats
TPC-C input
fields in such a manner that the keys can be extracted
in the above manner.
*/

```

```

int GetIntKeyValue(char **pQueryString, char *pKey,
WEBERROR NoKeyErr, WEBERROR NOTIntErr)
{
    char *ptr0;
    char *ptr;

    if ( !(ptr=strstr(*pQueryString, pKey)) )
        goto ErrorNoKey;
    ptr += strlen(pKey);
    if ( *ptr != '=' )
        goto ErrorNoKey;
    ptr++;

    ptr0 = ptr;          // remember
starting point
    // scan string until a terminator (null or &
or a non-digit
    while( *ptr && *ptr != '&'amp; && isdigit(*ptr) )
        ptr++;

    // make sure we stopped scanning for the
right reason
    if ((ptr0 == ptr) || (*ptr && *ptr != '&'))
    {
        if (NotIntErr != NO_ERR)
            throw new CWEBCLNT_ERR(
NoKeyErr );
        return 0;
    }

    *pQueryString = ptr;
    return atoi(ptr0);

ErrorNoKey:
    if (NoKeyErr != NO_ERR)
        throw new CWEBCLNT_ERR( NoKeyErr );
    return 0;
}

/* FUNCTION: TermInit
 * PURPOSE: This function initializes the
client terminal structure; it is called when the
TPCC.DLL
 * is first loaded by the
inet service.
 */
void TermInit(void)
{
    EnterCriticalSection(&TermCriticalSection);

    Term.iMasterSyncId = 1;
    Term.iNumEntries = Reg.dwMaxConnections+1;

    Term.pClientData = NULL;
    Term.pClientData =
(PCLIENTDATA)malloc(Term.iNumEntries *
sizeof(CLIENTDATA));
    if (Term.pClientData == NULL)
    {
        LeaveCriticalSection(&TermCriticalSection);
        throw new CWEBCLNT_ERR(
ERR_MEM_ALLOC_FAILED );
    }

    ZeroMemory( Term.pClientData,
Term.iNumEntries * sizeof(CLIENTDATA) );
}

```

```

Term.iFreeList =
Term.iNumEntries-1;
// build free list
// note: Term.pClientData[0].iNextFree gets
set to -1, which marks it as "in use".
// This is intentional, as the zero
entry is used as an anchor and never
// allocated as an actual
terminal.
for(int i=0; i<Term.iNumEntries; i++)
    Term.pClientData[i].iNextFree = i-1;

LeaveCriticalSection(&TermCriticalSection);

/* FUNCTION: TermDeleteAll
 * PURPOSE: This function frees allocated
resources associated with the terminal structure.
 * ARGUMENTS: none
 * RETURNS: None
 * COMMENTS: This function is called only when
the inet service unloads the TPCC.DLL
 */
void TermDeleteAll(void)
{
    EnterCriticalSection(&TermCriticalSection);
    for(int i=1; i<Term.iNumEntries; i++)
    {
        if (Term.pClientData[i].iNextFree
== -1)
            delete
Term.pClientData[i].pTxn;
    }

    Term.iFreeList = 0;
    Term.iNumEntries = 0;
    if ( Term.pClientData )
        free(Term.pClientData);
    Term.pClientData = NULL;

    LeaveCriticalSection(&TermCriticalSection);
}

/* FUNCTION: TermAdd
 * PURPOSE: This function assigns a terminal id
which is used to identify a client browser.
 * RETURNS: int
assigned terminal id
 */
int TermAdd(void)
{
    DWORD i;
    int iNewTerm, iTickCount;

    if (Term.iNumEntries == 0)
        return -1;

    EnterCriticalSection(&TermCriticalSection);
    if (Term.iFreeList != 0)
    {
        // position is available

```

```

iNewTerm = Term.iFreeList;
Term.iFreeList =
Term.pClientData[iNewTerm].iNextFree;
Term.pClientData[iNewTerm].iNextFree = -1;
// indicates this position is in use
}
else
{
    // no open slots, so find the slot
that hasn't been used in the longest time and reuse it
    for(iNewTerm=1, i=1,
iTickCount=0x7FFFFFFF; i<Reg.dwMaxConnections; i++)
    {
        if (iTickCount >
Term.pClientData[i].iTickCount)
        {
            iTickCount =
Term.pClientData[i].iTickCount;
            iNewTerm = i;
        }
    }
    // if oldest term is less than one
minute old, it probably means that more connections
// are being attempted than were
specified as "Max Connections" at install. In this
case,
// do not bump existing connection;
instead, return error to requestor.
    if ((GetTickCount() - iTickCount) <
60000)
    {
        LeaveCriticalSection(&TermCriticalSection);
        throw new CWEBCLNT_ERR(
ERR_MAX_CONNECTIONS_EXCEEDED );
    }

    Term.pClientData[iNewTerm].iTickCount =
GetTickCount();
    Term.pClientData[iNewTerm].iSyncId =
Term.iMasterSyncId++;
    Term.pClientData[iNewTerm].pTxn = NULL;

    LeaveCriticalSection(&TermCriticalSection);
    return iNewTerm;
}

/* FUNCTION: TermDelete
 * PURPOSE: This function makes a terminal
entry in the Term array available for reuse.
 * ARGUMENTS: int id
Terminal id of client exiting
 */
void TermDelete(int id)
{
    if ( id > 0 && id < Term.iNumEntries )
    {
        delete Term.pClientData[id].pTxn;
        // put onto free list

        EnterCriticalSection(&TermCriticalSection);

        Term.pClientData[id].iNextFree =
Term.iFreeList;
        Term.iFreeList = id;
    }
}

```

```

}
LeaveCriticalSection(&TermCriticalSection);
}
}

```

```

/* FUNCTION: MakeErrorForm
*/

```

```

void ErrorForm(EXTENSION_CONTROL_BLOCK *pECB, int
iType, int iErrorNum, int iTermId, int iSyncId, char
*szErrorText, char *szBuffer)
{

```

```

    wsprintf(szBuffer,
    "<HTML><HEAD><TITLE>TPC-C
Error</TITLE></HEAD><BODY>"
    "<FORM ACTION=\"tpcc.d11\"
METHOD=\"GET\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"STATUSID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"ERROR\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"TERMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
    "<BOLD>An Error
Occurred</BOLD><BR><BR>"
    "%s"
    "<BR><BR><HR>"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Payment..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Delivery..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Order-Status..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Exit..\">"
    "</FORM></BODY></HTML>"
    iType, iErrorNum, MAIN_MENU_FORM,
iTermId, iSyncId, szErrorText);
}

```

```

/* FUNCTION: MakeMainMenuForm
*/

```

```

void MakeMainMenuForm(int iTermId, int iSyncId, char
*szForm)
{
    wsprintf(szForm,
    "<HTML><HEAD><TITLE>TPC-C Main
Menu</TITLE></HEAD><BODY>"
    "select Desired
Transaction.<BR><HR>"
    "<FORM ACTION=\"tpcc.d11\"
METHOD=\"GET\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"STATUSID\" VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"ERROR\" VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"TERMID\" VALUE=\"%d\">"

```

```

    "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Payment..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Delivery..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Order-Status..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
    "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Exit..\">"
    "</FORM></BODY></HTML>"
    , MAIN_MENU_FORM, iTermId,
iSyncId);
}
/* FUNCTION: MakeStockLevelForm
*
* PURPOSE: This function constructs the Stock
Level HTML page.
*
* COMMENTS: The internal client buffer is
created when the terminal id is assigned and should not
* be freed except
when the client terminal id is no longer needed.
*/
void MakeStockLevelForm(int iTermId, STOCK_LEVEL_DATA
*pStockLevelData, BOOL bInput, char *szForm)
{
    int c;
    c = wsprintf(szForm,
    "<HTML><HEAD><TITLE>TPC-C Stock
Level</TITLE></HEAD><FORM ACTION=\"tpcc.d11\"
METHOD=\"GET\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"STATUSID\" VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"ERROR\" VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"TERMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
    "<PRE><font face=\"Courier\">
Stock-Level<BR>"
    "warehouse: %6.6d District:
%2.2d<BR><BR>"
    , STOCK_LEVEL_DATA, iTermId,
Term.pClientData[iTermId].iSyncId,
Term.pClientData[iTermId].w_id,
Term.pClientData[iTermId].d_id);
    if ( bInput )
    {
        strcpy(szForm+c,
        "Stock Level Threshold:
low stock:
</font><BR><BR><BR><BR><BR><BR><BR><BR><BR>
<BR><BR><BR></PRE><HR>"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Process\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Menu\">"

```

```

    "</FORM></HTML>" );
    }
    else
    {
        wsprintf(szForm+c,
        "Stock Level Threshold:
%2.2d<BR><BR>"
        "low stock: %3.3d</font>
<BR><BR><BR><BR><BR><BR><BR><BR><BR>
<BR><BR><BR><BR></PRE><HR>"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Payment..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Delivery..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Order-Status..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
        "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Exit..\">"
        "</FORM></HTML>"
        , pStockLevelData->low_stock);
    }
}
/* FUNCTION: MakeNewOrderForm
*
* COMMENTS: The internal client buffer is
created when the terminal id is assigned and should not
* be freed except
when the client terminal id is no longer needed.
*/
void MakeNewOrderForm(int iTermId, NEW_ORDER_DATA
*pNewOrderData, BOOL bInput, char *szForm)
{
    int i, c;
    BOOL bValid;
    static char szBR[] = "<BR><BR><BR><BR>
<BR><BR><BR><BR><BR><BR><BR><BR><BR>";
    if (!bInput)
        assert( (pNewOrderData->exec_status_code == eOK || pNewOrderData->exec_status_code == eInvalidItem) );
    bValid = (bInput || (pNewOrderData->exec_status_code == eOK));
    c = wsprintf(szForm,
    "<HTML><HEAD><TITLE>TPC-C New
Order</TITLE></HEAD><BODY>"
    "<FORM ACTION=\"tpcc.d11\"
METHOD=\"GET\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"STATUSID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"ERROR\" VALUE=\"0\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"FORMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"TERMID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"%d\">"
    "<PRE><font face=\"Courier\">
New Order<BR>"

```

```

, bValid ? 0 : ERR_BAD_ITEM_ID,
NEW_ORDER_FORM, iTermId,
Term.pClientData[iTermId].iSyncId);
    if ( bInput )
    {
        c += sprintf(szForm+c, "warehouse:
%6.6d ", Term.pClientData[iTermId].w_id );
        strcpy( szForm+c,
                "District: <INPUT
                Date:<BR>"
                "Customer: <INPUT
NAME=\"DID*\" SIZE=1"
NAME=\"CID*\" SIZE=4" Name:
Credit: %Disc:<BR>"
                "Order Number:
Number of Lines: W_tax: D_tax:<BR>"
                "Supp_W Item_Id Item
Name Qty Stock B/G Price
Amount<BR>"
                " <INPUT NAME=\"SP00*\"
                <INPUT NAME=\"IID00*\" SIZE=6"
                <INPUT NAME=\"Qty00*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP01*\"
                <INPUT NAME=\"IID01*\" SIZE=6"
                <INPUT NAME=\"Qty01*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP02*\"
                <INPUT NAME=\"IID02*\" SIZE=6"
                <INPUT NAME=\"Qty02*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP03*\"
                <INPUT NAME=\"IID03*\" SIZE=6"
                <INPUT NAME=\"Qty03*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP04*\"
                <INPUT NAME=\"IID04*\" SIZE=6"
                <INPUT NAME=\"Qty04*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP05*\"
                <INPUT NAME=\"IID05*\" SIZE=6"
                <INPUT NAME=\"Qty05*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP06*\"
                <INPUT NAME=\"IID06*\" SIZE=6"
                <INPUT NAME=\"Qty06*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP07*\"
                <INPUT NAME=\"IID07*\" SIZE=6"
                <INPUT NAME=\"Qty07*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP08*\"
                <INPUT NAME=\"IID08*\" SIZE=6"
                <INPUT NAME=\"Qty08*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP09*\"
                <INPUT NAME=\"IID09*\" SIZE=6"
                <INPUT NAME=\"Qty09*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP10*\"
                <INPUT NAME=\"IID10*\" SIZE=6"
                <INPUT NAME=\"Qty10*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP11*\"
                <INPUT NAME=\"IID11*\" SIZE=6"
                <INPUT NAME=\"Qty11*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP12*\"
                <INPUT NAME=\"IID12*\" SIZE=6"
                <INPUT NAME=\"Qty12*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP13*\"
                <INPUT NAME=\"IID13*\" SIZE=6"
                <INPUT NAME=\"Qty13*\" SIZE=1><BR>"
                " <INPUT NAME=\"SP14*\"
                <INPUT NAME=\"IID14*\" SIZE=6"
                <INPUT NAME=\"Qty14*\" SIZE=1><BR>"
                "Execution Status:
Total:<BR>"
                "</font></PRE><HR>"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"Process\">"

```

```

NAME=\"CMD\" VALUE=\"Menu\">" <INPUT TYPE=\"submit\"
" </FORM></HTML>"
);
    }
    else
    {
        c += sprintf(szForm+c, "warehouse:
%6.6d District: %2.2d Date: ",
                pNewOrderData->w_id,
                pNewOrderData->d_id);
        if ( bValid )
        {
            c += sprintf(szForm+c,
"%2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d",
                pNewOrderData-
>o_entry_d.day,
                pNewOrderData-
>o_entry_d.month,
                pNewOrderData-
>o_entry_d.year,
                pNewOrderData-
>o_entry_d.hour,
                pNewOrderData-
>o_entry_d.minute,
                pNewOrderData-
>o_entry_d.second);
        }
        c += sprintf(szForm+c,
"<BR>Customer: %4.4d Name: %-16s Credit: %-2s ",
                pNewOrderData->c_id,
                pNewOrderData->c_last, pNewOrderData->c_credit);
        if ( bValid )
        {
            c += sprintf(szForm+c,
                "%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
                B/G Price Amount<BR>",
                100.0*pNewOrderData->c_discount,
                pNewOrderData-
>o_id,
                pNewOrderData-
>o_o1_cnt,
                100.0 *
                pNewOrderData->w_tax,
                100.0 *
                pNewOrderData->d_tax);
        }
        for(i=0; i<pNewOrderData-
>o_o1_cnt; i++)
        {
            c +=
                sprintf(szForm+c, "%6.6d %6.6d %-24s %2.2d
                %3.3d %1.1s %6.2f %7.2f <BR>",
                pNewOrderData->oL[i].oL_supply_w_id,
                pNewOrderData->oL[i].oL_i_id,
                pNewOrderData->oL[i].oL_i_name,
                pNewOrderData->oL[i].oL_quantity,

```

```

                pNewOrderData->oL[i].oL_stock,
                pNewOrderData->oL[i].oL_brand_generic,
                pNewOrderData->oL[i].oL_i_price,
                pNewOrderData->oL[i].oL_amount );
        }
        else
        {
            c += sprintf(szForm+c,
                "%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>"
                pNewOrderData->o_id);
            i = 0;
        }
        strcpy( szForm+c, szBR, (15-i)*5
);
        c += (15-i)*5;
        if ( bValid )
        {
            c += sprintf(szForm+c,
                "Execution Status: Transaction committed.
                Total: $%8.2f ",
                pNewOrderData-
>total_amount);
        }
        else
        {
            c += sprintf(szForm+c,
                "Execution Status: Item number is not valid.
                Total:");
            strcpy(szForm+c,
                "<BR></font></PRE><HR>"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..NewOrder..\">"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Payment..\">"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Delivery..\">"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Order-Status..\">"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Stock-Level..\">"
                "<INPUT TYPE=\"submit\"
NAME=\"CMD\" VALUE=\"..Exit..\">"
                "</FORM></HTML>"
                );
        }
    }
}
/* FUNCTION: MakePaymentForm
*
* COMMENTS: The internal client buffer is
* created when the terminal id is assigned and should not
* be freed except
* when the client terminal id is no longer needed.
*/
void MakePaymentForm(int iTermId, PAYMENT_DATA
*pPaymentData, BOOL bInput, char *szForm)
{
    int c;

```



```

Order-Status<BR>" "<PRE><font face="Courier">
"Warehouse: %6.6d ",
ORDER_STATUS_FORM, iTermId,
Term.pClientData[iTermId].iSyncId,
Term.pClientData[iTermId].w_id;

if ( bInput )
{
strcpy(szForm+c,
"District: <INPUT
NAME="DID*" SIZE=1><BR>"
"Customer: <INPUT
NAME="CID*" SIZE=4> Name:
<INPUT NAME="CLT*" SIZE=23><BR>"
"Order-Number:
Entry-Date: Carrier-Number:<BR>"
"Supply-W Item-Id
Qty Amount Delivery-Date<BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
" <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
" <HR><INPUT
TYPE="submit" NAME="CMD" VALUE="Process"><INPUT
TYPE="submit" NAME="CMD" VALUE="Menu">"
"</BODY></FORM></HTML>"
);
}
else
{
c += sprintf(szForm+c,
"District: %2.2d<BR>"
"Customer: %4.4d Name:
pOrderStatusData->d_id,
pOrderStatusData->c_id,
pOrderStatusData->c_first, pOrderStatusData->c_middle, pOrderStatusData->c_last);

c += sprintf(szForm+c, "Cust-
Balance: %$9.2f<BR> <BR>",
pOrderStatusData->c_balance);

c += sprintf(szForm+c,
"Order-Number: %8.8d
Carrier-Number: %2.2d<BR>"
"Supply-W Item-Id
Qty Amount Delivery-Date<BR>",
pOrderStatusData->o_id,
pOrderStatusData->o_entry_d.day,
pOrderStatusData->o_entry_d.month,
pOrderStatusData->o_entry_d.year,
pOrderStatusData->o_entry_d.hour,
pOrderStatusData->o_entry_d.minute,
pOrderStatusData->o_entry_d.second,
pOrderStatusData->o_carrier_id);

for(i=0; i< pOrderStatusData->o_o1_cnt; i++)
{

```

```

c += sprintf(szForm+c, "
%6.6d %6.6d %2.2d %$8.2f %2.2d-%2.2d-
%4.4d<BR>",
pOrderStatusData->o1[i].o1_supply_w_id,
pOrderStatusData->o1[i].o1_id,
pOrderStatusData->o1[i].o1_quantity,
pOrderStatusData->o1[i].o1_amount,
pOrderStatusData->o1[i].o1_delivery_d.day,
pOrderStatusData->o1[i].o1_delivery_d.month,
pOrderStatusData->o1[i].o1_delivery_d.year);
}
strncpy( szForm+c, szBR, (15-i)*5
);
c += (15-i)*5;
strcpy(szForm+c,
" </font></PRE><HR><INPUT
TYPE="submit" NAME="CMD" VALUE="NewOrder..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Payment..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Delivery..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Order-Status..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Stock-Level..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Exit..">"
"</BODY></FORM></HTML>"
);
}
}
/* FUNCTION: MakeDeliveryForm
* COMMENTS: The internal client buffer is
created when the terminal id is assigned and should not
be freed except
when the client terminal id is no longer needed.
*/
void MakeDeliveryForm(int iTermId, DELIVERY_DATA
*pDeliveryData, BOOL bInput, char *szForm)
{
int c;

c = sprintf(szForm,
" <HTML><HEAD><TITLE>TPC-C
Delivery</TITLE></HEAD><BODY>"
" <FORM ACTION="tpcc.dll"
METHOD="GET">"
" <INPUT TYPE="hidden"
NAME="STATUSID" VALUE="%d">"
" <INPUT TYPE="hidden"
NAME="ERROR" VALUE="0">"
" <INPUT TYPE="hidden"
NAME="FORMID" VALUE="%d">"
" <INPUT TYPE="hidden"
NAME="TERMID" VALUE="%d">"
" <INPUT TYPE="hidden"
NAME="SYNCID" VALUE="%d">"
"<PRE><font face="Courier">
Delivery<BR>"
"Warehouse: %6.6d<BR> <BR>",

```

```

(!bInput && (pDeliveryData-
>exec_status_code != eOK)) ? ERR_TYPE_DELIVERY_POST :
0,
DELIVERY_FORM, iTermId,
Term.pClientData[iTermId].iSyncId,
Term.pClientData[iTermId].w_id);

if ( bInput )
{
strcpy( szForm+c,
"Carrier Number: <INPUT
NAME="OCD*" SIZE=1><BR> <BR>
"Execution Status: <BR>
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
" <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> </font></PRE><HR>"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="Process">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="Menu">"
"</BODY></FORM></HTML>"
);
}
else
{
wsprintf( szForm+c,
"Carrier Number:
%2.2d<BR> <BR>"
"Execution Status: %s
<BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR> </font></PRE>"
" <HR><INPUT
TYPE="submit" NAME="CMD" VALUE="NewOrder..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Payment..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Delivery..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Order-Status..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Stock-Level..">"
" <INPUT TYPE="submit"
NAME="CMD" VALUE="..Exit..">"
"</BODY></FORM></HTML>"
, pDeliveryData->o_carrier_id,
pDeliveryData->exec_status_code == eOK) ? "Delivery has been queued."
: "Delivery Post Failed "
);
}
}
/* FUNCTION: ProcessNewOrderForm
* PURPOSE: This function gets and validates
the input data from the new order form
* filling in the required
input variables. it then calls the SQLNewOrder
transaction, constructs
the output form and writes it back to client
browser.
*/
void ProcessNewOrderForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
PNEW_ORDER_DATA pNewOrder;

```

```

        pNewOrder = Term.pClientData[iTermId].pTxn-
>BuffAddr_NewOrder();

        ZeroMemory(pNewOrder,
sizeof(NEW_ORDER_DATA));
        pNewOrder->w_id =
Term.pClientData[iTermId].w_id;
        GetNewOrderData(pECB->lpszQueryString,
pNewOrder);

        Term.pClientData[iTermId].pTxn->NewOrder();

        pNewOrder = Term.pClientData[iTermId].pTxn-
>BuffAddr_NewOrder();
        MakeNewOrderForm(iTermId, pNewOrder,
OUTPUT_FORM, szBuffer );
}

/* FUNCTION: void ProcessPaymentForm
* PURPOSE:      This function gets and validates
the input data from the payment form
*              filling in the required
input variables. It then calls the SQLPayment
transaction, constructs
the output form and writes it back to client
browser.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB
               passed in structure pointer from inetsrv.
               int
               iTermId client browser terminal id
*/

void ProcessPaymentForm(EXTENSION_CONTROL_BLOCK *pECB,
int iTermId, char *szBuffer)
{
        PPAYMENT_DATA      pPayment;

        pPayment = Term.pClientData[iTermId].pTxn-
>BuffAddr_Payment();
        ZeroMemory(pPayment, sizeof(PAYMENT_DATA));
        pPayment->w_id =
Term.pClientData[iTermId].w_id;
        GetPaymentData(pECB->lpszQueryString,
pPayment);

        Term.pClientData[iTermId].pTxn->Payment();

        pPayment = Term.pClientData[iTermId].pTxn-
>BuffAddr_Payment();
        MakePaymentForm(iTermId, pPayment,
OUTPUT_FORM, szBuffer);
}

/* FUNCTION: ProcessOrderStatusForm
* PURPOSE:      This function gets and validates
the input data from the Order Status
*              form filling in the
required input variables. It then calls the
SQLOrderStatus
transaction, constructs the output form and writes it
back to client browser.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB
               passed in structure pointer from inetsrv.
               int
               iTermId client browser terminal id
*/

```

```

*
*/

void ProcessOrderStatusForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
        PORDER_STATUS_DATA pOrderStatus;

        pOrderStatus =
Term.pClientData[iTermId].pTxn->BuffAddr_OrderStatus();
        ZeroMemory(pOrderStatus,
sizeof(ORDER_STATUS_DATA));
        pOrderStatus->w_id =
Term.pClientData[iTermId].w_id;
        GetOrderStatusData(pECB->lpszQueryString,
pOrderStatus);

        Term.pClientData[iTermId].pTxn-
>OrderStatus();

        pOrderStatus =
Term.pClientData[iTermId].pTxn->BuffAddr_OrderStatus();
        MakeOrderStatusForm(iTermId, pOrderStatus,
OUTPUT_FORM, szBuffer);
}

/* FUNCTION: ProcessDeliveryForm
* PURPOSE:      This function gets and validates
the input data from the delivery form
*              filling in the required
input variables. It then calls the PostDeliveryInfo
*              Api. The client is then
informed that the transaction has been posted.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB
               passed in structure pointer from inetsrv.
               int
               iTermId client browser terminal id
*/

void ProcessDeliveryForm(EXTENSION_CONTROL_BLOCK *pECB,
int iTermId, char *szBuffer)
{
        char *ptr = pECB->lpszQueryString;

        PDELIVERY_DATA      pDelivery;

        pDelivery = Term.pClientData[iTermId].pTxn-
>BuffAddr_Delivery();
        ZeroMemory(pDelivery, sizeof(DELIVERY_DATA));
        pDelivery->w_id =
Term.pClientData[iTermId].w_id;

        pDelivery->o_carrier_id =
GetIntKeyValue(&ptr, "OCD*",
ERR_DELIVERY_MISSING_OCD_KEY,
ERR_DELIVERY_CARRIER_INVALID);
        if ( pDelivery->o_carrier_id > 10 ||
pDelivery->o_carrier_id < 1 )
                throw new CWEBCLNT_ERR(
ERR_DELIVERY_CARRIER_ID_RANGE );

        if (dwNumDeliveryThreads)
        {
                //post delivery info
                if ( PostDeliveryInfo(pDelivery-
>w_id, pDelivery->o_carrier_id ) )
                        pDelivery-
>exec_status_code = eDeliveryFailed;

```

```

                else
                        pDelivery-
>exec_status_code = eOK;
        }
        else // delivery is done synchronously if no
delivery threads configured
                Term.pClientData[iTermId].pTxn-
>Delivery();

        pDelivery = Term.pClientData[iTermId].pTxn-
>BuffAddr_Delivery();
        MakeDeliveryForm(iTermId, pDelivery,
OUTPUT_FORM, szBuffer);
}

/* FUNCTION: ProcessStockLevelForm
* PURPOSE:      This function gets and validates
the input data from the Stock Level
*              form filling in the
required input variables. It then calls the
SQLStockLevel
transaction, constructs the output form and writes it
back to client browser.
*
* ARGUMENTS:   EXTENSION_CONTROL_BLOCK *pECB
               passed in structure pointer from inetsrv.
               int
               iTermId client browser terminal id
*/

void ProcessStockLevelForm(EXTENSION_CONTROL_BLOCK
*pECB, int iTermId, char *szBuffer)
{
        char *ptr = pECB-
>lpszQueryString;

        PSTOCK_LEVEL_DATA   pStockLevel;

        pStockLevel = Term.pClientData[iTermId].pTxn-
>BuffAddr_StockLevel();
        ZeroMemory( pStockLevel,
sizeof(STOCK_LEVEL_DATA) );

        pStockLevel->w_id =
Term.pClientData[iTermId].w_id;
        pStockLevel->d_id =
Term.pClientData[iTermId].d_id;

        pStockLevel->threshold = GetIntKeyValue(&ptr,
"TT*", ERR_STOCKLEVEL_MISSING_THRESHOLD_KEY,
ERR_STOCKLEVEL_THRESHOLD_INVALID);
        if ( pStockLevel->threshold >= 100 ||
pStockLevel->threshold < 0 )
                throw new CWEBCLNT_ERR(
ERR_STOCKLEVEL_THRESHOLD_RANGE );

        Term.pClientData[iTermId].pTxn->StockLevel();

        pStockLevel = Term.pClientData[iTermId].pTxn-
>BuffAddr_StockLevel();
        MakeStockLevelForm(iTermId, pStockLevel,
OUTPUT_FORM, szBuffer);
}

/* FUNCTION: GetNewOrderData
* PURPOSE:      This function extracts and
validates the new order form data from an http command
string.

```



```

*
* ARGUMENTS:      LPSTR      client browser
                  lpszQueryString
http command string
*
                  NEW_ORDER_DATA
                  pointer to new
order data structure
*
*/

void GetNewOrderData(LPSTR lpszQueryString,
NEW_ORDER_DATA *pNewOrderData)
{
    char    szTmp[26];
    int     i;
    short   items;
    int     ol_i_id, ol_quantity;
    char    *ptr = lpszQueryString;

    static char szSP[MAX_OL_NEW_ORDER_ITEMS][6] =
"SP03*", "SP04*",
"SP08*", "SP09*",
"SP13*", "SP14*";
    static char szIID[MAX_OL_NEW_ORDER_ITEMS][7]
=
"IID03*", "IID04*",
"IID08*", "IID09*",
"IID13*", "IID14*";
    static char szQty[MAX_OL_NEW_ORDER_ITEMS][7]
=
"Qty03*", "Qty04*",
"Qty08*", "Qty09*",
"Qty13*", "Qty14*";

    pNewOrderData->d_id = GetIntKeyValue(&ptr,
"DID*", ERR_NEWORDER_FORM_MISSING_DID,
ERR_NEWORDER_DISTRICT_INVALID);
    pNewOrderData->c_id = GetIntKeyValue(&ptr,
"CID*", ERR_NEWORDER_CUSTOMER_KEY,
ERR_NEWORDER_CUSTOMER_INVALID);
    for(i=0, items=0; i<MAX_OL_NEW_ORDER_ITEMS;
i++)
    {
        GetKeyValue(&ptr, szSP[i], szTmp,
sizeof(szTmp), ERR_NEWORDER_MISSING_SUPPW_KEY);
        if ( szTmp[0] )
            if ( !IsNumeric(szTmp) )
                throw new
CWEBCLNT_ERR( ERR_NEWORDER_SUPPW_INVALID );
        pNewOrderData->
>OL[items].ol_supply_w_id = atoi(szTmp);

        ol_i_id = pNewOrderData->
>OL[items].ol_i_id =
            GetIntKeyValue(&ptr, szIID[i],
ERR_NEWORDER_MISSING_IID_KEY,
ERR_NEWORDER_ITEMID_INVALID);
            if ( ol_i_id > 999999 ||
ol_i_id < 1 )

```

```

                throw new
CWEBCLNT_ERR( ERR_NEWORDER_ITEMID_RANGE );
            ol_quantity =
pNewOrderData->OL[items].ol_quantity =
                GetIntKeyValue(&ptr, szQty[i],
ERR_NEWORDER_MISSING_QTY_KEY,
ERR_NEWORDER_QTY_INVALID);
                if ( ol_quantity > 99 ||
ol_quantity < 1 )
                    throw new
CWEBCLNT_ERR( ERR_NEWORDER_QTY_RANGE );
                items++;
            }
            else
                // nothing entered for
supply warehouse, so item id and qty must also be blank
                GetKeyValue(&ptr,
szIID[i], szTmp, sizeof(szTmp),
ERR_NEWORDER_MISSING_IID_KEY);
                if ( szTmp[0] )
                    throw new
CWEBCLNT_ERR( ERR_NEWORDER_ITEMID_WITHOUT_SUPPW );
                GetKeyValue(&ptr,
szQty[i], szTmp, sizeof(szTmp),
ERR_NEWORDER_MISSING_QTY_KEY);
                if ( szTmp[0] )
                    throw new
CWEBCLNT_ERR( ERR_NEWORDER_QTY_WITHOUT_SUPPW );
            }
            if ( items == 0 )
                throw new CWEBCLNT_ERR(
ERR_NEWORDER_NOITEMS_ENTERED );
        }
        pNewOrderData->o_ol_cnt = items;
    }

/* FUNCTION: GetPaymentData
*
* PURPOSE:      This function extracts and
validates the payment form data from an http command
string.
*
* ARGUMENTS:   LPSTR      client browser
                  lpszQueryString
http command string
                  PAYMENT_DATA
                  pointer to
                  *pPaymentData
                  payment data structure
*/

void GetPaymentData(LPSTR lpszQueryString, PAYMENT_DATA
*pPaymentData)
{
    char    szTmp[26];
    char    *ptr = lpszQueryString;
    BOOL    bCustIdBlank;

    pPaymentData->d_id = GetIntKeyValue(&ptr,
"DID*", ERR_PAYMENT_MISSING_DID_KEY,
ERR_PAYMENT_DISTRICT_INVALID);
    GetKeyValue(&ptr, "CID*", szTmp,
sizeof(szTmp), ERR_PAYMENT_MISSING_CID_KEY);
    if ( szTmp[0] == 0 )
        {
            bCustIdBlank = TRUE;
            pPaymentData->c_id = 0;

```

```

        }
    }
    else
    {
        // parse customer id and verify
        that last name was NOT entered
        bCustIdBlank = FALSE;
        if ( !IsNumeric(szTmp) )
            throw new CWEBCLNT_ERR(
ERR_PAYMENT_CUSTOMER_INVALID );
        pPaymentData->c_id = atoi(szTmp);
    }

    pPaymentData->c_w_id = GetIntKeyValue(&ptr,
"CWI*", ERR_PAYMENT_MISSING_CWI_KEY,
ERR_PAYMENT_CWI_INVALID);
    pPaymentData->c_d_id = GetIntKeyValue(&ptr,
"CDI*", ERR_PAYMENT_MISSING_CDI_KEY,
ERR_PAYMENT_CDI_INVALID);
    if ( bCustIdBlank )
        // customer id is blank, so last
name must be entered
        GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_PAYMENT_MISSING_CLT_KEY);
        if ( szTmp[0] == 0 )
            throw new CWEBCLNT_ERR(
ERR_PAYMENT_MISSING_CID_CLT );
        _strupr( szTmp );
        if ( strlen(pPaymentData->c_last) >
LAST_NAME_LEN )
            throw new CWEBCLNT_ERR(
ERR_PAYMENT_LAST_NAME_TO_LONG );
        strcpy(pPaymentData->c_last,
szTmp);
    }
    else
    {
        // parse customer id and verify
        that last name was NOT entered
        GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_PAYMENT_MISSING_CLT_KEY);
        if ( szTmp[0] != 0 )
            throw new CWEBCLNT_ERR(
ERR_PAYMENT_CID_AND_CLT );
    }

    GetKeyValue(&ptr, "HAM*", szTmp,
sizeof(szTmp), ERR_PAYMENT_MISSING_HAM_KEY);
    if ( !IsDecimal(szTmp) )
        throw new CWEBCLNT_ERR(
ERR_PAYMENT_HAM_INVALID );
    pPaymentData->h_amount = atof(szTmp);
    if ( pPaymentData->h_amount >= 10000.00 ||
pPaymentData->h_amount < 0 )
        throw new CWEBCLNT_ERR(
ERR_PAYMENT_HAM_RANGE );
}

/* FUNCTION: GetOrderStatusData
*
* PURPOSE:      This function extracts and
validates the payment form data from an http command
string.
*/

void GetOrderStatusData(LPSTR lpszQueryString,
ORDER_STATUS_DATA *pOrderStatusData)
{
    char    szTmp[26];
    char    *ptr = lpszQueryString;

```

```

        pOrderStatusData->d_id = GetIntKeyValue(&ptr,
"DIR*", ERR_ORDERSTATUS_MISSING_DIR_KEY,
ERR_ORDERSTATUS_DIR_INVALID);

        GetKeyValue(&ptr, "CID*", szTmp,
sizeof(szTmp), ERR_ORDERSTATUS_MISSING_CID_KEY);
        if ( szTmp[0] == 0 )
        {
            // customer id is blank, so last
            name must be entered
            pOrderStatusData->c_id = 0;
            GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_ORDERSTATUS_MISSING_CLT_KEY);
            if ( szTmp[0] == 0 )
                throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_MISSING_CID_CLT );

            _strupr( szTmp );
            if ( strlen(pOrderStatusData-
>c_last) > LAST_NAME_LEN )
                throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CLT_RANGE );
            strcpy(pOrderStatusData->c_last,
szTmp);
        }
        else
        {
            // parse customer id and verify
            that last name was NOT entered
            if ( !IsNumeric(szTmp) )
                throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CID_INVALID );
            pOrderStatusData->c_id =
atoi(szTmp);
            GetKeyValue(&ptr, "CLT*", szTmp,
sizeof(szTmp), ERR_ORDERSTATUS_MISSING_CLT_KEY);
            if ( szTmp[0] != 0 )
                throw new CWEBCLNT_ERR(
ERR_ORDERSTATUS_CID_AND_CLT );
        }
    }

/* FUNCTION: BOOL IsNumeric(char *ptr)
* PURPOSE: This function determines if a
string is numeric. It fails if any characters other
than numeric and null
terminator are present.
* ARGUMENTS: char *ptr
pointer to string to check.
* RETURNS: BOOL FALSE if
string is not all numeric
TRUE if string contains only numeric
characters i.e. '0' - '9'
*/
BOOL IsNumeric(char *ptr)
{
    if ( *ptr == 0 )
        return FALSE;

    while( *ptr && isdigit(*ptr) )
        ptr++;

    return ( !*ptr );
}

/* FUNCTION: BOOL IsDecimal(char *ptr)
* PURPOSE: This function determines if a
string is a non-negative decimal value.

```

```

* It fails if any characters other than a
series of numbers followed by a decimal point, another
series of numbers, and a null terminator are present.
* ARGUMENTS: char *ptr
pointer to string to check.
* RETURNS: BOOL FALSE if
string is not a valid non-negative decimal value
TRUE if string is OK
*/
BOOL IsDecimal(char *ptr)
{
    char *dotptr;
    BOOL bvalid;

    if ( *ptr == 0 )
        return FALSE;

    // find decimal point
    dotptr = strchr( ptr, '.' );
    if (dotptr == NULL)
        // no decimal point, so just check
        return IsNumeric(ptr);
    // temporarily replace decimal
    with a terminator
    if ( *ptr != 0 )
        bvalid = IsNumeric(ptr);
    // string starts with decimal point
    else if (*(dotptr+1) == 0)
        return FALSE; // nothing but a
    decimal point is bad
    else
        bvalid = TRUE;

    if (*(dotptr+1) != 0)
        // check text after decimal point
        bvalid &= IsNumeric(dotptr+1);

    *dotptr = '.'; // replace decimal point
    return bvalid;
}

isapi_dll/src/resource.h
//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by tpcc.rc
//
#define IDD_DIALOG1 101

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE 102
#define _APS_NEXT_COMMAND_VALUE 40001
#define _APS_NEXT_CONTROL_VALUE 1000
#define _APS_NEXT_SYMED_VALUE 101
#endif
#endif

```

common/src/ReadRegistry.cpp

```

/* FILE: READREGISTRY.CPP
Microsoft TPC-C
Kit Ver. 4.20.000
Copyright
Microsoft, 1999
All Rights Reserved
not yet audited
PURPOSE: Implementation for TPC-C Tuxedo
Class.
Contact: Charles Levine
(clevine@microsoft.com)
Change history:
4.20.000 - first version
*/

/* FUNCTION: ReadTPCCRegistrySettings
* PURPOSE: This function reads the NT registry
for startup parameters. There parameters are
under the TPCC key.
* RETURNS FALSE = no errors
TRUE = error reading
registry
*/
BOOL ReadTPCCRegistrySettings( TPCCREGISTRYDATA *pReg )
{
    HKEY hkey;
    DWORD size;
    DWORD type;
    DWORD dwTmp;
    char szTmp[256];

    if ( RegOpenKeyEx(HKEY_LOCAL_MACHINE,
"SOFTWARE\\Microsoft\\TPCC", 0, KEY_READ, &hkey) !=
ERROR_SUCCESS )
        return TRUE;

    // determine database protocol to use; may be
    either ODBC or DBLIB
    pReg->eDB_Protocol = Unspecified;
    size = sizeof(szTmp);
    if ( RegQueryValueEx(hkey, "DB_Protocol", 0,
&type, (BYTE *)szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp,
szDBNames[ODBC] ) )
            pReg->eDB_Protocol =
ODBC;
        else if ( !strcmp(szTmp,
szDBNames[DBLIB] ) )
            pReg->eDB_Protocol =
DBLIB;
    }

    pReg->eTxnMon = None;
    // determine txn monitor to use; may be
    either TUXEDO, or blank
    size = sizeof(szTmp);
    if ( RegQueryValueEx(hkey, "TxnMonitor", 0,
&type, (BYTE *)szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp,
szTxnMonNames[TUXEDO] ) )

```

```

        else if ( !strcmp(szTmp,
szTxnMonNames[ENCINA]) )
            pReg->eTxnMon = TUXEDO;
        else if ( !strcmp(szTmp,
szTxnMonNames[COM]) )
            pReg->eTxnMon = COM;
    }

    pReg->bCOM_SinglePool = FALSE;
    size = sizeof(szTmp);
    if ( RegQueryValueEx(hkey, "COM_SinglePool",
0, &type, (BYTE *)&szTmp, &size) == ERROR_SUCCESS )
    {
        if ( !strcmp(szTmp, "YES") )
            pReg->bCOM_SinglePool =
TRUE;
    }

    pReg->dwMaxConnections = 0;
    size = sizeof(dwTmp);
    if ( ( RegQueryValueEx(hkey,
"MaxConnections", 0, &type, (LPBYTE)&dwTmp, &size) ==
ERROR_SUCCESS )
        && (type == REG_DWORD) )
        pReg->dwMaxConnections = dwTmp;

    pReg->dwMaxPendingDeliveries = 0;
    size = sizeof(dwTmp);
    if ( ( RegQueryValueEx(hkey,
"MaxPendingDeliveries", 0, &type, (LPBYTE)&dwTmp,
&size) == ERROR_SUCCESS )
        && (type == REG_DWORD) )
        pReg->dwMaxPendingDeliveries =
dwTmp;

    pReg->dwNumberOfDeliveryThreads = 0;
    size = sizeof(dwTmp);
    if ( ( RegQueryValueEx(hkey,
"NumberOfDeliveryThreads", 0, &type, (LPBYTE)&dwTmp,
&size) == ERROR_SUCCESS )
        && (type == REG_DWORD) )
        pReg->dwNumberOfDeliveryThreads =
dwTmp;

    size = sizeof( pReg->szPath );
    if ( RegQueryValueEx(hkey, "Path", 0, &type,
(BYTE *)&pReg->szPath, &size) != ERROR_SUCCESS )
        pReg->szPath[0] = 0;

    size = sizeof( pReg->szDbServer );
    if ( RegQueryValueEx(hkey, "DbServer", 0,
&type, (BYTE *)&pReg->szDbServer, &size) !=
ERROR_SUCCESS )
        pReg->szDbServer[0] = 0;

    size = sizeof( pReg->szDbName );
    if ( RegQueryValueEx(hkey, "DbName", 0,
&type, (BYTE *)&pReg->szDbName, &size) != ERROR_SUCCESS
)
        pReg->szDbName[0] = 0;

    size = sizeof( pReg->szDbUser );
    if ( RegQueryValueEx(hkey, "DbUser", 0,
&type, (BYTE *)&pReg->szDbUser, &size) != ERROR_SUCCESS
)
        pReg->szDbUser[0] = 0;

    size = sizeof( pReg->szDbPassword );
    if ( RegQueryValueEx(hkey, "DbPassword", 0,
&type, (BYTE *)&pReg->szDbPassword, &size) !=
ERROR_SUCCESS )

```

```

        pReg->szDbPassword[0] = 0;

    size = sizeof( pReg->szSPPrefix );
    if ( RegQueryValueEx(hkey, L"SPPrefix", 0,
&type, (BYTE *)&pReg->szSPPrefix, &size) !=
ERROR_SUCCESS )
        pReg->szSPPrefix[0] = L'\0';

    RegCloseKey(hkey);

    return FALSE;
}

```

common/src/ReadRegistry.h

```

/* FILE: ReadRegistry.h
Microsoft TPC-C
Kit Ver. 4.20.000
Copyright
Microsoft, 1999
All Rights Reserved
not audited
PURPOSE: Header for registry related code.
Change history:
4.20.000 - first version
*/

enum DBPROTOCOL { Unspecified, ODBC, DBLIB };
const char *szDBNames[] = { "Unspecified", "ODBC",
"DBLIB" };

enum TXNMON { None, TUXEDO, ENCINA, COM };
const char *szTxnMonNames[] = { "NONE", "TUXEDO",
"ENCINA", "COM" };

//This structure defines the data necessary to keep
distinct for each terminal or client connection.
typedef struct _TPCCREGISTRYDATA
{
    enum DBPROTOCOL eDb_Protocol;
    enum TXNMON eTxnMon;
    BOOL bCOM_SinglePool;
    DWORD dwMaxConnections;
    DWORD dwMaxPendingDeliveries;
    DWORD dwNumberOfDeliveryThreads;
    char szPath[128];
    char szDbServer[32];
    char szDbName[32];
    char szDbUser[32];
    char szDbPassword[32];
    wchar_t szSPPrefix[32]; //tpcc_odbc.d11
    stored procedures prefix
} TPCCREGISTRYDATA, *PTPCCREGISTRYDATA;

BOOL ReadTPCCRegistrySettings( TPCCREGISTRYDATA *pReg
);

```

common/src/error.h

```

/* FILE: ERROR.H
Microsoft TPC-C
Kit Ver. 4.20.000
Copyright
Microsoft, 1999
All Rights Reserved
Version
4.10.000 audited by Richard Gimarc, Performance
Metrics, 3/17/99
PURPOSE: Header file for error exception
classes.
Change history:
4.20.000 - updated rev number to
match kit
4.21.000 - fixed bug: ~CBaseErr
needed to be declared virtual
*/

#pragma once

#ifndef _INC_STRING
#include <string.h>
#endif

const int m_szMsg_size = 512;
const int m_szApp_size = 64;
const int m_szLoc_size = 64;

//error message structure used in ErrorText routines
typedef struct _SERRORMSG
{
    int iError; //error id of message
    char szMsg[256]; //message to sent to browser
} SERRORMSG;

typedef enum _ErrorLevel
{
    ERR_FATAL_LEVEL = 2, = 1,
    ERR_WARNING_LEVEL = 3,
    ERR_INFORMATION_LEVEL
} ErrorLevel;

#define ERR_TYPE_LOGIC -1 //logic error in program; internal error
#define ERR_SUCCESS 0 //success (a non-error error)
#define ERR_BAD_ITEM_ID 1 //expected abort record in txnRecord
#define ERR_TYPE_DELIVERY_POST 2 //expected delivery post failed
#define ERR_TYPE_WEBDLL 3 //tpcc web generated error
#define ERR_TYPE_SQL 4 //sql server generated error
#define ERR_TYPE_DBLIB 5 //dblib generated error
#define ERR_TYPE_ODBC 6 //odbc generated error

```

```

#define ERR_TYPE_SOCKET 7
//error on communication socket client rte
only
#define ERR_TYPE_DEADLOCK 8
//dblib and odbc only deadlock condition
#define ERR_TYPE_COM 9
//error from COM call
#define ERR_TYPE_TUXEDO 10
//tuxedo error
#define ERR_TYPE_OS 11
//operating system error
#define ERR_TYPE_MEMORY 12
//memory allocation error
#define ERR_TYPE_TPCC_ODBC 13
//error from tpcc odbc txn module
#define ERR_TYPE_TPCC_DBLIB 14
//error from tpcc dblib txn module
#define ERR_TYPE_DELISRV 15
//delivery server error
#define ERR_TYPE_TXNLOG 16
//txn log error
#define ERR_TYPE_BCCONN 17
//Benchcraft connection class
#define ERR_TYPE_TPCC_CONN 18
//Benchcraft connection class
#define ERR_TYPE_ENCINA 19
//Encina error
#define ERR_TYPE_COMPONENT 20
//error from COM component
#define ERR_TYPE_RTE 21
//Benchcraft rte
#define ERR_TYPE_AUTOMATION 22
//Benchcraft automation errors
#define ERR_TYPE_DRIVER 23
//Driver engine errors
#define ERR_TYPE_RTE_BASE 24
//Framework errors
#define ERR_BUF_OVERFLOW 25
//Buffer overflow during receive
#define ERR_TYPE_SOAP_HTTP 26
//HTTP/SOAP dll generated error
// TPC-W error types
#define ERR_TYPE_TPCW_CONN 50
//Benchcraft connection class
#define ERR_TYPE_TPCW_HTML 51
//error from TpcwHtml dll
#define ERR_TYPE_TPCW_USER 52
//error from TPC-W user class

```

```

#define ERR_TYPE_TPCW_ENG_BASE 53
#define ERR_TYPE_TPCW_ENG_OS 54
#define ERR_TYPE_HTML_RESP 55
#define ERR_TYPE_TPCW_ODBC 56
#define ERR_TYPE_SCHANNEL 57
#define ERR_TYPE_THINK_LIST 58

#define ERR_INS_MEMORY "Insufficient Memory to continue."
#define ERR_UNKNOWN "Unknown error."
#define ERR_MSG_BUF_SIZE 512
#define INV_ERROR_CODE -1
#define ERR_INS_BUF_OVERFLOW "Insufficient Buffer size to receive HTML pages."

class CBaseErr
{
public:
    CBaseErr(LPCTSTR szLoc = NULL)
    {
        m_idMsg =
        GetLastError(); //take the error code immediately
        before it is reset by other functions

        if (szLoc)
        {
            m_szLoc = new
            char[strlen(szLoc)+1/*m_szLoc_size*/];
            strcpy(m_szLoc, szLoc);
        }
        else
            m_szLoc = NULL;

        m_szApp = new
        char[m_szApp_size];

        GetModuleFileName(GetModuleHandle(NULL),
        m_szApp, m_szApp_size);
    }

    CBaseErr(int idMsg, LPCTSTR szLoc = NULL)
    {
        m_idMsg = idMsg;

        if (szLoc)
        {
            m_szLoc = new
            char[strlen(szLoc)+1/*m_szLoc_size*/];
            strcpy(m_szLoc, szLoc);
        }
        else
            m_szLoc = NULL;

        m_szApp = new
        char[m_szApp_size];
    }
}

```

```

    GetModuleFileName(GetModuleHandle(NULL),
    m_szApp, m_szApp_size);
}

virtual ~CBaseErr(void)
{
    if (m_szApp)
        delete [] m_szApp;
    if (m_szLoc)
        delete [] m_szLoc;
};

virtual void Draw(HWND hwnd, LPCTSTR szStr =
NULL)
{
    int j = 0;
    char szTmp[512];
    if (szStr)
        j = wsprintf(szTmp,
"%s\n",szStr);
    if (ErrorNum() != INV_ERROR_CODE)
        j += wsprintf(szTmp+j,
"Error = %d\n", ErrorNum());
    if (m_szLoc)
        j += wsprintf(szTmp+j,
"Location = %s\n", GetLocation());
    ErrorText();
    j += wsprintf(szTmp+j, "%s\n",
::MessageBox(hwnd, szTmp, m_szApp,
MB_OK);
}

char *GetApp(void) { return m_szApp; }
char *GetLocation(void) { return m_szLoc; }
virtual int ErrorNum() { return m_idMsg; }

virtual int ErrorType() = 0; // a value
which distinguishes the kind of error that occurred
virtual char *ErrorText() = 0; // a string
(i.e., human readable) representation of the error

protected:
char *m_szApp;
char *m_szLoc; // code location where
the error occurred
int m_idMsg;

//short m_errType;
};

class CSocketErr : public CBaseErr
{
public:
    enum Action
    {
        eNone = 0,
        eSend,
        eSocket,
        eBind,
        eConnect,
        eListen,
        eHost,
        eRecv,
        eGetHostByName,
        eWSACreateEvent,
    }
}

```

```

        eWSASend,
        eWSAGetOverlappedResult,
        eWSARecv,
        eWSAWaitForMultipleEvents,
        eWSAStartup,
        eWSAResetEvent,
        eNonRetryable,
    };

    CSocketErr(Action eAction, LPCTSTR szLocation
= NULL);

~CSocketErr()
{
    if (m_szErrorText != NULL)
        delete [] m_szErrorText;
};

Action m_eAction;
char m_szErrorText;

int ErrorType() { return ERR_TYPE_SOCKET;};
char *ErrorText(void);

};

class CSystemErr : public CBaseErr
{
public:
    enum Action
    {
        eNone = 0,
        eTransactNamedPipe,
        eWaitNamedPipe,
        eSetNamedPipeHandleState,
        eCreateFile,
        eCreateProcess,
        eCallNamedPipe,
        eCreateEvent,
        eCreateThread,
        eVirtualAlloc,
        eReadFile = 10,
        eWriteFile,
        eMapViewOfFile,
        eCreateFileMapping,
        eInitializeSecurityDescriptor,
        eSetSecurityDescriptorDacl,
        eCreateNamedPipe,
        eConnectNamedPipe,
        eWaitForSingleObject,
        eRegOpenKeyEx,
        eRegQueryValueEx = 20,
        eBeginThread,
        eRegEnumValue,
        eRegSetValueEx,
        eRegCreateKeyEx,
        eWaitForMultipleObjects,
        eRegisterClassEx,
        eCreateWindow,
        eCreateSemaphore,
        eReleaseSemaphore,
        eFSeek,
        eFRead,
        eFWrite,
        eTmpFile,
        eSetFilePointer,
        eNew,
        eCloseHandle,
    };

    CSystemErr(Action
eAction, LPCTSTR szLocation);

```

```

        CSystemErr(int iError,
Action eAction, LPCTSTR szLocation);
int ErrorType() { return
ERR_TYPE_OS;};
char *ErrorText(void);
void Draw(HWND hwnd, LPCTSTR szStr =
NULL);

Action m_eAction;

private:
char m_szMsg[ERR_MSG_BUF_SIZE];
};

class CMemoryErr : public CBaseErr
{
public:
    CMemoryErr();

    int ErrorType() {return ERR_TYPE_MEMORY;};
    char *ErrorText() {return ERR_INS_MEMORY;};
};

class CBufferOverflowErr : public CBaseErr
{
public:
    CBufferOverflowErr(int,LPCTSTR);

    int ErrorType() {return ERR_BUF_OVERFLOW;};
    char *ErrorText() {return
ERR_INS_BUF_OVERFLOW;};
};

```

common/src/trans.h

```

/* FILE: TRANS.H Microsoft TPC-C
 * Kit Ver. 4.42.000 Copyright
 * Microsoft, 2002 All Rights Reserved
 * Version
 * 4.10.000 audited by Richard Gimarc, Performance
 * Metrics, 3/17/99
 *
 * PURPOSE: Header file for TPC-C structure
 * templates.
 *
 * Change history:
 * 4.42.000 - changed w_id fields from
 * short to long to support >32K warehouses
 * 4.20.000 - updated rev number to
 * match kit
 */
#pragma once

// String length constants
#define SERVER_NAME_LEN 20
#define DATABASE_NAME_LEN 20
#define USER_NAME_LEN 20
#define PASSWORD_LEN 20
#define TABLE_NAME_LEN 20
#define I_DATA_LEN 50
#define I_NAME_LEN 24

```

```

#define BRAND_LEN 1
#define LAST_NAME_LEN 16
#define W_NAME_LEN 10
#define ADDRESS_LEN 20
#define STATE_LEN 2
#define ZIP_LEN 9
#define S_DIST_LEN 24
#define S_DATA_LEN 50
#define D_NAME_LEN 10
#define FIRST_NAME_LEN 16
#define MIDDLE_NAME_LEN 2
#define PHONE_LEN 16
#define DATETIME_LEN 30
#define CREDIT_LEN 2
#define C_DATA_LEN 250
#define H_DATA_LEN 24
#define DIST_INFO_LEN 24
#define MAX_OL_NEW_ORDER_ITEMS 15
#define MAX_OL_ORDER_STATUS_ITEMS 15
#define STATUS_LEN 25
#define OL_DIST_INFO_LEN 24

// TIMESTAMP_STRUCT is provided by the ODBC header file
// sqltypes.h, but is not available
// when compiling with dllib, so redefined here. Note:
// we are using the symbol "SQLTYPES"
// (declared in sqltypes.h) as a way to determine if
// TIMESTAMP_STRUCT has been declared.
#ifndef SQLTYPES
typedef struct
{
    /* SQLSMALLINT */ short
    year; unsigned short /*
SQLSMALLINT */ month; unsigned short /*
SQLSMALLINT */ day; unsigned short /*
SQLSMALLINT */ hour; unsigned short /*
SQLSMALLINT */ minute; unsigned short /*
SQLSMALLINT */ second; unsigned long /*
SQLINTEGER */ fraction;
} TIMESTAMP_STRUCT;
#endif

// possible values for exec_status_code after
// transaction completes
enum EXEC_STATUS
{
    eOK, // 0
    "Transaction committed." // 0
    eInvalidItem, // 1 "Item number is
not valid."
    eDeliveryFailed // 2 "Delivery Post
Failed."
};

// transaction structures
typedef struct
{
    // input params
    long
    ol_supply_w_id;
    long
    ol_i_id;
    short
    ol_quantity;

    // output params

```

```

char
ol_i_name[I_NAME_LEN+1];
char
ol_brand_generic[BRAND_LEN+1];
double
ol_i_price;
double
ol_amount;
short
ol_stock;
} OL_NEW_ORDER_DATA;

typedef struct
{
    // input params
    long w_id;
    short d_id;
    long c_id;
    short o_ol_cnt;

    // output params
    EXEC_STATUS exec_status_code;
    char c_last[LAST_NAME_LEN+1];
    char c_credit[CREDIT_LEN+1];
    double c_discount;
    double w_tax;
    double d_tax;
    long o_id;
    short o_commit_flag;
    TIMESTAMP_STRUCT o_entry_d;
    short o_all_local;
    double total_amount;
    OL_NEW_ORDER_DATA OL_NEW_ORDER_DATA;
    OL[MAX_OL_NEW_ORDER_ITEMS];
} NEW_ORDER_DATA, *PNEW_ORDER_DATA;

typedef struct
{
    // input params
    long w_id;
    short d_id;
    long c_id;
    short c_d_id;
    long c_w_id;
    double h_amount;
    char c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS exec_status_code;
    TIMESTAMP_STRUCT h_date;
    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];
    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];
char
c_first[FIRST_NAME_LEN+1];
char
c_middle[MIDDLE_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];
TIMESTAMP_STRUCT c_since;
char
c_credit[CREDIT_LEN+1];
double
c_credit_lim;
double
c_discount;
double
c_balance;
char
c_data[200+1];
} PAYMENT_DATA, *PPAYMENT_DATA;

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

typedef struct
{
    // input params
    long w_id;
    short d_id;
    long c_id;
    char c_last[LAST_NAME_LEN+1];

    // output params
    EXEC_STATUS exec_status_code;
    char c_first[FIRST_NAME_LEN+1];
    char c_middle[MIDDLE_NAME_LEN+1];
    double c_balance;
    long o_id;
    long TIMESTAMP_STRUCT o_entry_d;
    short o_carrier_id;
    OL_ORDER_STATUS_DATA OL_ORDER_STATUS_DATA;
    OL[MAX_OL_ORDER_STATUS_ITEMS];
    short o_ol_cnt;
} ORDER_STATUS_DATA, *PORDER_STATUS_DATA;

```

```

typedef struct
{
    // input params
    long w_id;
    short o_carrier_id;

    // output params
    EXEC_STATUS exec_status_code;
    SYSTEMTIME queue_time;
    long o_id[10]; // id's of delivered
    orders for districts 1 to 10
} DELIVERY_DATA, *PDELIVERY_DATA;

//This structure is used for posting delivery
transactions and for writing them to the delivery
server.
typedef struct _DELIVERY_TRANSACTION
{
    SYSTEMTIME queue;
    //time delivery transaction queued
    long w_id;
    //delivery warehouse
    short o_carrier_id;
    //carrier id
} DELIVERY_TRANSACTION;

typedef struct
{
    // input params
    long w_id;
    short d_id;
    short threshold;

    // output params
    EXEC_STATUS exec_status_code;
    long low_stock;
} STOCK_LEVEL_DATA, *PSTOCK_LEVEL_DATA;

```

common/src/txn_base.h

```

/* FILE: TXN_BASE.H
 * Microsoft TPC-C
 * Copyright
 * Microsoft, 1999 All Rights Reserved
 * Version
 * 4.10.000 audited by Richard Gimarc, Performance
 * Metrics, 3/17/99
 * PURPOSE: Header file for TPC-C txn class
 * implementation.
 * Change history:
 * 4.20.000 - updated rev number to
 * match kit
 */
#pragma once

// need to declare functions for import, unless define
has already been created

```

```

// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class DllDecl CTPCC_BASE
{
public:
    CTPCC_BASE(void) {};
    virtual ~CTPCC_BASE(void) {};

    virtual PNEW_ORDER_DATA
    BuffAddr_NewOrder() = 0;
    virtual PPAYMENT_DATA
    BuffAddr_Payment() = 0;
    virtual PDELIVERY_DATA
    BuffAddr_Delivery() = 0;
    virtual PSTOCK_LEVEL_DATA
    BuffAddr_StockLevel() = 0;
    virtual PORDER_STATUS_DATA
    BuffAddr_OrderStatus() = 0;

    virtual void NewOrder
    () = 0;
    virtual void Payment
    () = 0;
    virtual void Delivery
    () = 0;
    virtual void StockLevel
    () = 0;
    virtual void OrderStatus
    () = 0;
};

```

db_dblib_dll/src/tpcc_dblib.cpp

```

/* FILE: TPCC_DBLIB.CPP
 * Microsoft TPC-C
 * Copyright
 * All Rights Reserved
 * Version
 * 4.10.000 audited by Richard Gimarc, Performance
 * Metrics, 3/17/99
 * PURPOSE: Implements dblib calls for TPC-C
 * txns. Contact: Charles Levine
 * (clevine@microsoft.com)
 * Change history:
 * 4.42.000 - changed w_id fields from
 * short to long to support >32k warehouses
 * 4.20.000 - updated rev number to
 * match kit
 * 4.10.001 - not deleting error class
 * in catch handler on deadlock retry;
 * not a
 * functional bug, but a memory leak
 * - had to tweak
 * some declarations to compile with latest SDK; no
 * functional change
 */
#include <windows.h>

```

```

#include <stdio.h>
#include <assert.h>

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

#ifdef ICECAP
#include <icapexp.h>
#endif

// need to declare functions for export
#define DllDecl __declspec( dllexport )

#include "..\..\common\src\error.h"
#include "..\..\common\src\trans.h"
#include "..\..\common\src\txn_base.h"
#include "tpcc_dblib.h"

#define DEFCLPACKSIZE 4096

// version string; must match return value from
tpcc_version stored proc
const char sVersion[] = "4.10.000";

const iMaxRetries = 10;
static long iConnectionCount = 0; // number of
current dblib connections

const int iErrOleDbProvider = 7312;
const char sErrTimeoutExpired[] = "Timeout expired";

BOOL WINAPIENTRY DllMain(HMODULE hModule, DWORD
ul_reason_for_call, LPVOID lpReserved)
{
    switch( ul_reason_for_call )
    {
        case DLL_PROCESS_ATTACH:
            DisableThreadLibraryCalls(hModule);
            dbinit(); //
            initialize dblib break;

        case DLL_PROCESS_DETACH:
            dbexit(); //
            close all dblib structures/connections
            break;

        default: // /* nothing */;
    }
    return TRUE;
}

int err_handler(DBPROCESS *dbproc, int severity, int
dberr, int oserr, LPCSTR dberrstr, LPCSTR oserrstr)
{
    CTPCC_DBLIB
    *pConn;
    assert(dbproc != NULL);
    pConn = (CTPCC_DBLIB*)dbgetuserdata(dbproc);

    if (pConn != NULL)
    {
        pConn->SetDblibError( severity,
        dberr, oserr, dberrstr, oserrstr );
    }
    return INT_CANCEL;
}

```

```

}

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

// typedef INT (SQLAPI *DBMSGHANDLE_PROC)(PDBPROCESS,
DBINT, INT, INT, LPCSTR, LPCSTR, LPCSTR, DBUSMALLINT);

int msg_handler(DBPROCESS *dbproc, DBINT msgno, int
msgstate, int severity, LPCSTR srvname, LPCSTR procname, DBUSMALLINT line)
{
    CTPCC_DBLIB
    *pConn;
    assert(dbproc != NULL);
    pConn = (CTPCC_DBLIB*)dbgetuserdata(dbproc);

    if (pConn != NULL)
    {
        pConn->SetSqlError( msgno,
        msgstate, severity, msgtext );
    }

    return 0;
}

/* FUNCTION: void UtilStrCpy(char *pDest, char *pSrc,
int n)
 *
 * PURPOSE: This function copies n characters
from string pSrc to pDest and places a
null character at the end
 *
 * ARGUMENTS: char destination string pointer
 * *pDest char
 * *pSrc source string pointer

```

```

*
*          n          int
*          number of characters to copy
* RETURNS:          None
* COMMENTS:        Unlike strncpy this function
ensures that the result string is          always null
terminated.
*/
inline static void UtilStrCpy(char * pDest, const BYTE
* pSrc, int n)
{
    strncpy(pDest, (char *)pSrc, n);
    pDest[n] = '\0';
}
return;
}
/* FUNCTION: CTPCC_DBLIB_ERR::ErrorText
*/
char* CTPCC_DBLIB_ERR::ErrorText(void)
{
    int i;
    static SERRORMSG errorMsgs[] =
    {
        { ERR_WRONG_SP_VERSION,
server" },
        { ERR_INVALID_CUST,
        "Invalid Customer id,name."
    },
        { ERR_NO_SUCH_ORDER,
        "No orders found for customer."
    },
        { ERR_RETRIED_TRANS,
        "Retries before transaction succeeded."
    },
        { 0,
        ""
    }
    };
    static char szNotFound[] = "Unknown error
number.";
    for(i=0; errorMsgs[i].szMsg[0]; i++)
    {
        if ( m_errno == errorMsgs[i].iError
)
            break;
    }
    if ( !errorMsgs[i].szMsg[0] )
        return szNotFound;
    else
        return errorMsgs[i].szMsg;
}
// wrapper routine for class constructor
_declspec(dllexport) CTPCC_DBLIB* CTPCC_DBLIB_new(
LPCSTR szServer,          // name of SQL
server
LPCSTR szUser,          //
user name for login

```

```

login
LPCSTR szPassword,          // password for
LPCSTR szHost,          //
workstation name; shows up in sp_who; max 30 chars,
only first 10 kept by SQL Server
LPCSTR szDatabase )          // name of
database to use
{
    return new CTPCC_DBLIB( szServer, szUser,
szPassword, szHost, szDatabase );
}
CTPCC_DBLIB::CTPCC_DBLIB (
LPCSTR szServer,          // name of SQL
server
LPCSTR szUser,          //
user name for login
LPCSTR szPassword,          // password for
login
LPCSTR szHost,          //
workstation name; shows up in sp_who; max 30 chars,
only first 10 kept by SQL Server
LPCSTR szDatabase )          // name of
database to use
{
    LOGINREC *login;
    const BYTE *pData;
    // initialization
    m_dbproc = NULL;
    m_DBLIBErr = (CDBLIBERR*)NULL;
    m_SqlErr = (CSQLERR*)NULL;
    m_MaxRetries = 10;          // how many
retries on
deadlock
    // increase max number of connections if
getting close
    if ( dbgetmaxprocs() < (iConnectionCount+5) )
    {
        if (
dbsetmaxprocs(iConnectionCount+10) == FAIL )
            ThrowError(CDBLIBERR::edbSetMaxProcs);
    }
    // allocate a login structure
    login = dblogin();
    if (login == NULL)
        ThrowError(CDBLIBERR::eLogin);
    InterlockedIncrement( &iConnectionCount );
    // register error and message handler
functions
    if (dbprocerrhandle(login, err_handler) ==
NULL)
        ThrowError(CDBLIBERR::edbProcHandler);
    if (dbprocsghandle(login, msg_handler) ==
NULL)
        ThrowError(CDBLIBERR::edbProcHandler);
    DBSETLUSER(login, szUser);
    DBSETLPWD(login, szPassword);
    DBSETLHOST(login, szHost);
    DBSETLPACKET(login, (unsigned
short)DEFCLPACKSIZE);

```

```

DBSETLVERSION(login, DBVER60);
// use dblink ver 6.0 client behavior
// set time to wait for login
if (dbsetlogintime(60) == FAIL)
    ThrowError(CDBLIBERR::edbSet);
// set time to wait for statement execution
if (dbsettime(180) == FAIL)
    ThrowError(CDBLIBERR::edbSet);
m_dbproc = dbopen(login, szServer);
// deallocate login structure before checking
for success
dbfreelogin( login );
if (m_dbproc == NULL)
    ThrowError(CDBLIBERR::edbOpen);
// save address of class instance so that the
message and error handler
// can get to data.
dbsetuserdata(m_dbproc, (LPVOID)this);
// Use the the right database
if (dbuse(m_dbproc, szDatabase) == FAIL)
    ThrowError(CDBLIBERR::edbUse);
dbcmd(m_dbproc, "set nocount on ");
// do not return row counts
dbcmd(m_dbproc, "set XACT_ABORT ON");
// rollback transaction on abort
if (dbsqlxexec(m_dbproc) == FAIL)
    ThrowError(CDBLIBERR::edbSqlExec);
DiscardNextResults(2);
// verify that version of stored procs on
server is correct
dbrpcinit(m_dbproc, "tpcc_version", 0);
if (dbrpcexec(m_dbproc) == FAIL)
    ThrowError(CDBLIBERR::edbRpcExec);
if (dbresults(m_dbproc) != SUCCEEDED)
    ThrowError(CDBLIBERR::edbResults);
if (dbnextrow(m_dbproc) != REG_ROW)
    ThrowError(CDBLIBERR::edbNextRow);
char szSrvVersion[16];
pData=dbdata(m_dbproc, 1);
if (pData)
    UtilStrCpy(szSrvVersion, pData,
dbdatlen(m_dbproc, 1));
else
    szSrvVersion[0]=0;
if (strcmp(szSrvVersion,sVersion))
    throw new CTPCC_DBLIB_ERR(
CTPCC_DBLIB_ERR::ERR_WRONG_SP_VERSION );
DiscardNextRows(0);
DiscardNextResults(0);
}
CTPCC_DBLIB::~CTPCC_DBLIB( void )
{

```



```

resources // close db connection and deallocate
dbcClose(m_dbproc);
InterlockedDecrement( &iConnectionCount );
if (m_DbLibErr != NULL)
    delete m_DbLibErr;
if (m_SqlErr != NULL)
    delete m_SqlErr;
}

void CTPCC_DBLIB::SetDbLibError(int severity, int
dberr, int oserr, LPCSTR dberrstr, LPCSTR oserrstr)
{
    delete m_DbLibErr;
    m_DbLibErr = new
CDBLIBERR(CDBLIBERR::eUnknown, severity, dberr, oserr);
    if (dberrstr != NULL)
    {
        m_DbLibErr->m_dberrstr = new char[
strlen(dberrstr)+1 ];
        strcpy( m_DbLibErr->m_dberrstr,
dberrstr );
    }
    if (oserrstr != NULL)
    {
        m_DbLibErr->m_oserrstr = new char[
strlen(oserrstr)+1 ];
        strcpy( m_DbLibErr->m_oserrstr,
oserrstr );
    }
}

void CTPCC_DBLIB::SetSqlError( int /*DBINT*/ msgno, int
msgstate, int severity, LPCSTR msgtext )
{
    if (m_SqlErr == NULL)
        m_SqlErr = new CSQLERR();

    m_SqlErr->m_msgno = msgno;
    m_SqlErr->m_msgstate = msgstate;
    m_SqlErr->m_severity = severity;

    delete [] m_SqlErr->m_msgtext;
    if (msgtext != NULL)
    {
        m_SqlErr->m_msgtext = new char[
strlen(msgtext)+1 ];
        strcpy( m_SqlErr->m_msgtext,
msgtext );
    }
}

void CTPCC_DBLIB::ThrowError( CDBLIBERR::ACTION eAction )
{
    // discard anything still in return buffer
    DiscardNextRows(-1);
    DiscardNextResults(-1);

    // check for SQL server error first; if yes,
    throw it and ignore any DBLib error.
    if (m_SqlErr != NULL)
    {
        CSQLERR *pSqlErr;
        pSqlErr = m_SqlErr;
        m_SqlErr = NULL; // clear our
        pointer to instance; catch handler will delete
        throw pSqlErr;
    }
}

```

```

CDBLIBERR *pDbLibErr;
if (m_DbLibErr == NULL)
    // this case isn't expected to
    happen, since it means that an error was returned
    // but the error handlers were not
    called.
    pDbLibErr = new CDBLIBERR(eAction);
else
{
    pDbLibErr = m_DbLibErr;
    pDbLibErr->m_eAction = eAction;
    m_DbLibErr = NULL; //
clear our pointer to instance; catch handler will
delete
}
throw pDbLibErr;

// Read and discard rows until no more. Throw an
exception if number of rows read doesn't
// match number of rows expected. The row count will
be ignored if the expected count value
// passed in is negative. A typical use of this
routine is to verify that there are no more
// rows to be read.
void CTPCC_DBLIB::DiscardNextRows(int iExpectedCount)
{
    int rc;
    RETCODE rc;
    while (TRUE)
    {
        rc = dbnextrow(m_dbproc);
        if (rc == NO_MORE_ROWS)
            break;
        if (rc == FAIL)
        {
            if (iExpectedCount >= 0)
                ThrowError(CDBLIBERR::eDbNextRow);
            else
                break;
        }
        iRowsRead++;
    }
    if ((iExpectedCount >= 0) &&
(iExpectedCount != iRowsRead))
        ThrowError(CDBLIBERR::eWrongRowCount);

// Read and discard results until no more. Throw an
exception if number of result sets read doesn't
// match number expected. The result set count will be
ignored if the expected count value
// passed in is negative. A typical use of this
routine is to verify that there are no more
// result sets to be read.
void CTPCC_DBLIB::DiscardNextResults(int
iExpectedCount)
{
    int rc;
    RETCODE rc;
    while (TRUE)
    {
        rc = dbresults(m_dbproc);
        if (rc == NO_MORE_RESULTS)
            break;
    }
}

```

```

if (rc == FAIL)
{
    if (iExpectedCount >= 0)
        ThrowError(CDBLIBERR::eDbResults);
    else
        break;
}
DiscardNextRows(-1);
iResultsRead++;
}
if ((iExpectedCount >= 0) &&
(iExpectedCount != iResultsRead))
    ThrowError(CDBLIBERR::eWrongRowCount);
}

void CTPCC_DBLIB::StockLevel()
{
    int iTryCount = 0;
    const BYTE *pData;
    ResetError();
    while (TRUE)
    {
        try
        {
            dbrpcinit(m_dbproc,
"tpcc_stocklevel", 0);
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.StockLevel.w_id); // @w_id int
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.StockLevel.d_id); // @d_id
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT2, -1, -1, (BYTE *)
&m_txn.StockLevel.threshold); // @threshold smallint
            if (dbrpcexec(m_dbproc)
== FAIL)
                ThrowError(CDBLIBERR::eDbRpcExec);
            if (dbresults(m_dbproc)
!= SUCCEEDED)
                ThrowError(CDBLIBERR::eDbResults);
            if (dbnextrow(m_dbproc)
!= REG_ROW)
                ThrowError(CDBLIBERR::eDbNextRow);
            if
(pData=dbdata(m_dbproc, 1))
                m_txn.StockLevel.low_stock = *((long *)
pData);
            DiscardNextRows(0);
            DiscardNextResults(0);
            m_txn.StockLevel.exec_status_code = eOK;
            return;
        }
    }
}

```



```

(pData=dbdata(m_dbproc, 11)) if
    UtilStrCpy(m_txn.Payment.d_city, pData,
dbdatlen(m_dbproc, 11));
    if
(pData=dbdata(m_dbproc, 12))
        UtilStrCpy(m_txn.Payment.d_state, pData,
dbdatlen(m_dbproc, 12));
    if
(pData=dbdata(m_dbproc, 13))
        UtilStrCpy(m_txn.Payment.d_zip, pData,
dbdatlen(m_dbproc, 13));
    if
(pData=dbdata(m_dbproc, 14))
        UtilStrCpy(m_txn.Payment.c_first, pData,
dbdatlen(m_dbproc, 14));
    if
(pData=dbdata(m_dbproc, 15))
        UtilStrCpy(m_txn.Payment.c_middle, pData,
dbdatlen(m_dbproc, 15));
    if
(pData=dbdata(m_dbproc, 16))
        UtilStrCpy(m_txn.Payment.c_street_1, pData,
dbdatlen(m_dbproc, 16));
    if
(pData=dbdata(m_dbproc, 17))
        UtilStrCpy(m_txn.Payment.c_street_2, pData,
dbdatlen(m_dbproc, 17));
    if
(pData=dbdata(m_dbproc, 18))
        UtilStrCpy(m_txn.Payment.c_city, pData,
dbdatlen(m_dbproc, 18));
    if
(pData=dbdata(m_dbproc, 19))
        UtilStrCpy(m_txn.Payment.c_state, pData,
dbdatlen(m_dbproc, 19));
    if
(pData=dbdata(m_dbproc, 20))
        UtilStrCpy(m_txn.Payment.c_zip, pData,
dbdatlen(m_dbproc, 20));
    if
(pData=dbdata(m_dbproc, 21))
        UtilStrCpy(m_txn.Payment.c_phone, pData,
dbdatlen(m_dbproc, 21));
    if
(pData=dbdata(m_dbproc, 22))
    {
        datetime =
*((DBDATETIME *) pData);
        dbdatecrack(m_dbproc, &daterec, &datetime);
        m_txn.Payment.c_since.year = daterec.year;
        m_txn.Payment.c_since.month = daterec.month;
        m_txn.Payment.c_since.day = daterec.day;
        m_txn.Payment.c_since.hour = daterec.hour;

```

```

        m_txn.Payment.c_since.minute =
daterec.minute;
        m_txn.Payment.c_since.second =
daterec.second;
    }
    if(pData=dbdata(m_dbproc,
23))
        UtilStrCpy(m_txn.Payment.c_credit, pData,
dbdatlen(m_dbproc, 23));
    if(pData=dbdata(m_dbproc,
24))
        dbconvert(m_dbproc, SQLNUMERIC,
(LPCBYTE)pData, dbdatlen(m_dbproc,24), SQLFLT8, (BYTE
*)&m_txn.Payment.c_credit_lim, 8);
    if(pData=dbdata(m_dbproc,
25))
        dbconvert(m_dbproc, SQLNUMERIC,
(LPCBYTE)pData, dbdatlen(m_dbproc,25), SQLFLT8, (BYTE
*)&m_txn.Payment.c_discount, 8);
    if(pData=dbdata(m_dbproc,
26))
        dbconvert(m_dbproc, SQLNUMERIC,
(LPCBYTE)pData, dbdatlen(m_dbproc,26), SQLFLT8, (BYTE
*)&m_txn.Payment.c_balance, 8);
    if(pData=dbdata(m_dbproc,
27))
        UtilStrCpy(m_txn.Payment.c_data, pData,
dbdatlen(m_dbproc, 27));
        DiscardNextRows(0);
        DiscardNextResults(0);
    if (m_txn.Payment.c_id ==
0)
        throw new
CTPCC_DBLIB_ERR( CTPCC_DBLIB_ERR::ERR_INVALID_CUST );
        else
            m_txn.Payment.exec_status_code = eOK;
            return;
        }
        catch (CSQLERR *e)
        {
            if ((e->m_msgno == 1205
||
(e->m_msgno ==
iErrorDbProvider &&
strchr(e-
>m_msgtext, sErrTimeoutExpired) != NULL)) &&
(++iTryCount <=
iMaxRetries))
            {
                // hit
                // deadlock; backoff for increasingly longer period
                delete e;
                sleep(10 * *
iTryCount);
            }
            else
                throw;
        }
    } // while (TRUE)

```

```

// if (iTryCount)
// throw new
CTPCC_DBLIB_ERR(CTPCC_DBLIB_ERR::ERR_RETRIED_TRANS,
iTryCount);
}

void CTPCC_DBLIB::OrderStatus()
{
    int
    DBDATETIME      datetime;
    DBDATEREK      daterec;

    int
    RETCODE         rc;
    const BYTE      *pData;

    ResetError();

    while (TRUE)
    {
        try
        {
            dbrpcinit(m_dbproc,
"tpcc_orderstatus", 0);
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.OrderStatus.w_id);
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.OrderStatus.d_id);
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.OrderStatus.c_id);

            // if customer id is
            // name
            // if
            zero, then order status is by
            (m_txn.OrderStatus.c_id == 0)

            dbrpcparam(m_dbproc, NULL, 0, SQLCHAR, -1,
strlen(m_txn.OrderStatus.c_last), (unsigned char
*)&m_txn.OrderStatus.c_last);

            if (dbrpcexec(m_dbproc)
== FAIL)
                ThrowError(CDBLIBERR::eDbRpcExec);

            // Get order lines
            // if (dbresults(m_dbproc)
!= SUCCEED)
            {
                if ((m_DbLibErr
== NULL) && (m_SqlErr == NULL))
                    throw
                    new CTPCC_DBLIB_ERR( CTPCC_DBLIB_ERR::ERR_NO_SUCH_ORDER
);
                else
                    ThrowError(CDBLIBERR::eDbResults);
            }

            if (dbnumcols(m_dbproc)
!= 5)
                ThrowError(CDBLIBERR::eWrongNumCols);

            i = 0;

```

```

while (TRUE)
{
    rc =
    if (rc ==
NO_MORE_ROWS)
        break;
    if (rc !=
REG_ROW)
        ThrowError(CDBLIBERR::edbNextRow);

    if(pData=dbdata(m_dbproc, 1))
        m_txn.OrderStatus.OL[i].ol_supply_w_id =
        (*DBSMALLINT *) pData);
    if(pData=dbdata(m_dbproc, 2))
        m_txn.OrderStatus.OL[i].ol_i_id = (*DBINT *)
pData);
    if(pData=dbdata(m_dbproc, 3))
        m_txn.OrderStatus.OL[i].ol_quantity =
        (*DBSMALLINT *) pData);
    if(pData=dbdata(m_dbproc, 4))
        dbconvert(m_dbproc, SQLNUMERIC,
(LPCBYTE)pData, dbdatlen(m_dbproc,4),
SQLFLT8, (BYTE
*)&m_txn.OrderStatus.OL[i].ol_amount, 8);
    if(pData=dbdata(m_dbproc, 5)) {
        datetime = *((DBDATETIME *) pData);
        dbdatecrack(m_dbproc, &daterec, &datetime);
        m_txn.OrderStatus.OL[i].ol_delivery_d.year
= daterec.year;
        m_txn.OrderStatus.OL[i].ol_delivery_d.month
= daterec.month;
        m_txn.OrderStatus.OL[i].ol_delivery_d.day
= daterec.day;
        m_txn.OrderStatus.OL[i].ol_delivery_d.hour
= daterec.hour;
        m_txn.OrderStatus.OL[i].ol_delivery_d.minute
= daterec.minute;
        m_txn.OrderStatus.OL[i].ol_delivery_d.second
= daterec.second;
    }
    i++;
    m_txn.OrderStatus.o_ol_cnt = i;

    if (dbresults(m_dbproc)
!= SUCCEED)
        ThrowError(CDBLIBERR::edbResults);
}

```

```

if (dbnextrow(m_dbproc)
!= REG_ROW)
    ThrowError(CDBLIBERR::edbNextRow);
if (dbnumcols(m_dbproc)
!= 8)
    ThrowError(CDBLIBERR::ewrongNumCols);
if(pData=dbdata(m_dbproc,
1))
    m_txn.OrderStatus.c_id = (*DBINT *) pData);
if(pData=dbdata(m_dbproc,
2))
    UtilStrCpy(m_txn.OrderStatus.c_last, pData,
dbdatlen(m_dbproc,2));
if(pData=dbdata(m_dbproc,
3))
    UtilStrCpy(m_txn.OrderStatus.c_first, pData,
dbdatlen(m_dbproc,3));
if(pData=dbdata(m_dbproc,
4))
    UtilStrCpy(m_txn.OrderStatus.c_middle, pData,
dbdatlen(m_dbproc, 4));
if(pData=dbdata(m_dbproc,
5))
    {
        datetime =
        *((DBDATETIME *) pData);
        dbdatecrack(m_dbproc, &daterec, &datetime);
        m_txn.OrderStatus.o_entry_d.year =
daterec.year;
        m_txn.OrderStatus.o_entry_d.month =
daterec.month;
        m_txn.OrderStatus.o_entry_d.day =
daterec.day;
        m_txn.OrderStatus.o_entry_d.hour =
daterec.hour;
        m_txn.OrderStatus.o_entry_d.minute =
daterec.minute;
        m_txn.OrderStatus.o_entry_d.second =
daterec.second;
    }
if(pData=dbdata(m_dbproc,
6))
    m_txn.OrderStatus.o_carrier_id =
        (*DBSMALLINT *) pData);
if(pData=dbdata(m_dbproc,
7))
    dbconvert(m_dbproc, SQLNUMERIC,
(LPCBYTE)pData, dbdatlen(m_dbproc,7),
SQLFLT8, (BYTE
*)&m_txn.OrderStatus.c_balance, 8);
if(pData=dbdata(m_dbproc,
8))

```

```

m_txn.OrderStatus.o_id = (*DBINT *) pData);
        DiscardNextRows(0);
        DiscardNextResults(0);
    }
    if
(m_txn.OrderStatus.o_ol_cnt == 0)
        throw new
CTPCC_DBLIB_ERR( CTPCC_DBLIB_ERR::ERR_NO_SUCH_ORDER );
    else if
(m_txn.OrderStatus.c_id == 0 &&
m_txn.OrderStatus.c_last[0] == 0)
        throw new
CTPCC_DBLIB_ERR( CTPCC_DBLIB_ERR::ERR_INVALID_CUST );
    else
        m_txn.OrderStatus.exec_status_code = eOK;
    return;
}
catch (CSQLERR *e)
{
    if ((e->m_msgno == 1205
||
(e->m_msgno ==
iErrorDbProvider &&
str(e->m_msgtext, serrTimeoutExpired) != NULL)) &&
(++iTryCount <=
iMaxRetries))
    {
        // hit
        deadlock; backoff for increasingly longer period
        delete e;
        Sleep(10 *
iTryCount);
    }
    else
        throw;
}
// while (TRUE)
// if (iTryCount)
// throw new
CTPCC_DBLIB_ERR(CTPCC_DBLIB_ERR::ERR_RETRIED_TRANS,
iTryCount);
}

void CTPCC_DBLIB::Delivery()
{
    int i;
    int iTryCount = 0;
    const BYTE *pData;
    ResetError();
    while (TRUE)
    {
        try
        {
            dbrpcinit(m_dbproc,
"tpcc_delivery", 0);
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.Delivery_w_id);
            dbrpcparam(m_dbproc,
NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.Delivery_o_carrier_id);

```

```

        if (dbrpcexec(m_dbproc)
== FAIL)
    ThrowError(CDBLIBERR::edbRpcExec);
        if (dbresults(m_dbproc)
!= SUCCEEDED)
    ThrowError(CDBLIBERR::edbResults);
        if (dbnextrow(m_dbproc)
!= REG_ROW)
    ThrowError(CDBLIBERR::edbNextRow);
        if (dbnumcols(m_dbproc)
!= 10)
    ThrowError(CDBLIBERR::ewrongNumCols);
        for (i=0; i<10; i++)
        {
            if (pData =
dbdata(m_dbproc, i+1))
                m_txn.Delivery.o_id[i] = *((DBINT *)pData);
            DiscardNextRows(0);
            DiscardNextResults(0);

            m_txn.Delivery.exec_status_code = eOK;
            return;
        }
        catch (CSQLERR *e)
        {
            if ((e->m_msgno == 1205
||
iErr0leDbProvider &&
>m_msgtext, sErrTimeoutExpired) != NULL) &&
iMaxRetries))
                {
                    // hit
                    deadlock; backoff for increasingly longer period
                    delete e;
                    Sleep(10 *
iTryCount);
                }
            else
                throw;
        }
    } // while (TRUE)
//
// if (iTryCount)
// throw new
CTPCC_DBLIB_ERR(CTPCC_DBLIB_ERR::ERR_RETRIED_TRANS,
iTryCount);
}

void CTPCC_DBLIB::ResetError()
{
    if (m_DbLibErr != NULL)
    {
        delete m_DbLibErr;
        m_DbLibErr = (CDBLIBERR*)NULL;
    }
}

```

```

        if (m_SqlErr != NULL)
        {
            delete m_SqlErr;
            m_SqlErr = (CSQLERR*)NULL;
        }
    }
    return;
}

db_dblib_dll/src/tpcc_dblib.h

/* FILE: TPCC_DBLIB.H
 * Microsoft TPC-C
 * Kit Ver. 4.20.000
 * Copyright
 * Microsoft, 1999
 * All Rights Reserved
 * Version
 * 4.10.000 audited by Richard Gimarc, Performance
 * Metrics, 3/17/99
 * PURPOSE: Header file for TPC-C txn class
 * implementation.
 * change history: 4.20.000 - updated rev number to
 * match kit
 */
#pragma once

#ifdef PDBPROCESS
#define DBPROCESS void // dbprocess structure type
typedef DBPROCESS * PDBPROCESS;
#endif

// need to declare functions for import, unless define
has already been created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class CSQLERR : public CBaseErr
{
public:
    CSQLERR(void)
    {
        m_msgno = 0;
        m_msgstate = 0;
        m_severity = 0;
        m_msgtext = NULL;
    };
    ~CSQLERR()
    {
        delete [] m_msgtext;
    };
    int m_msgno;
    int m_msgstate;
    int m_severity;
    char *m_msgtext;
    int ErrorType() {return
ERR_TYPE_SQL;};
    int ErrorNum() {return m_msgno;};
}

```

```

        char *ErrorText() {return
m_msgtext;};
};

class CDBLIBERR : public CBaseErr
{
public:
    enum ACTION
    {
        eNone,
        eunknown,
        eLogin,
        // error from dblogin
        eDbOpen,
        // error from dbopen
        eDbuse,
        // error from dbuse
        eDbSqlExec,
        // error from dbsqlExec,
        eDbSet,
        // error from dbset,
        // error from one of the dbset*
        eDbNextRow,
        // error from dbnextrow
        eWrongRowCount,
        // more or less rows returned than expected
        eWrongNumCols,
        // more or less columns returned than
        eDbResults,
        // error from dbresults
        eDbRpcExec,
        // error from dbrpcExec,
        eDbSetMaxProcs,
        // error from dbsetmaxprocs,
        eDbProcHandler,
        // error from either dbprocerrhandle or
        dbprocsghandle
    };
    CDBLIBERR(ACTION eAction, int
severity = 0, int dberror = 0, int oserr = 0)
    {
        m_eAction = eAction;
        m_severity = severity;
        m_dberror = dberror;
        m_oserr = oserr;

        m_dberrstr = NULL;
        m_oserrstr = NULL;
    };
    ~CDBLIBERR()
    {
        delete [] m_dberrstr;
        delete [] m_oserrstr;
    };
    ACTION m_eAction;
    int m_severity;
    int m_dberror;
    int m_oserr;
    char *m_dberrstr;
    char *m_oserrstr;
    int ErrorType() {return
ERR_TYPE_DBLIB;};
    int ErrorNum() {return m_dberror;};
    char *ErrorText() {return
m_dberrstr;};
}

```

```

};
class CTPCC_DBLIB_ERR : public CBaseErr
{
public:
    enum CTPCC_DBLIB_ERRS
    {
        ERR_WRONG_SP_VERSION = 1,
server" // "Wrong version of stored procs on database
        ERR_INVALID_CUST,
        // "Invalid customer id.name."
        ERR_NO_SUCH_ORDER,
        // "No orders found for customer."
        ERR_RETRIED_TRANS,
        // "Retries before transaction
succeeded."
    };
    CTPCC_DBLIB_ERR( int iErr ) {
m_errno = iErr; m_iTryCount = 0; };
    CTPCC_DBLIB_ERR( int iErr, int
iTryCount ) { m_errno = iErr; m_iTryCount = iTryCount;
};

    int m_errno;
    int m_iTryCount;

    int ErrorType() {return
ERR_TYPE_TPCC_DBLIB;};
    int ErrorNum() {return m_errno;};
    char *ErrorText();
};

class DllDecl CTPCC_DBLIB : public CTPCC_BASE
{
private:
    // declare variables and private
functions here...
    PDBPROCESS m_dbproc;
    CDBLIBERR *m_DbLibErr;
    // not allocated until needed (maybe never)
    CSQLERR *m_SqlErr;
    // not allocated until
needed (maybe never)
    int m_MaxRetries; // retry count
on deadlock

    void DiscardNextRows(int
iExpectedCount);
    void DiscardNextResults(int
iExpectedCount);
    void ThrowError( CDBLIBERR::ACTION
eAction );
    void ResetError();

    union
    {
        NEW_ORDER_DATA
NewOrder;
        PAYMENT_DATA
Payment;
        DELIVERY_DATA
Delivery;
        STOCK_LEVEL_DATA
StockLevel;
        ORDER_STATUS_DATA
OrderStatus;
    };
};

```

```

}
    m_txn;
public:
    CTPCC_DBLIB(LPCSTR szServer, LPCSTR
szUser, LPCSTR szPassword, LPCSTR szHost, LPCSTR
szDatabase );
    ~CTPCC_DBLIB(void);

    inline PNEW_ORDER_DATA
BuffAddr_NewOrder() { return
&m_txn.NewOrder; };
    inline PPAYMENT_DATA
BuffAddr_Payment() { return
&m_txn.Payment; };
    inline PDELIVERY_DATA
BuffAddr_Delivery() { return
&m_txn.Delivery; };
    inline PSTOCK_LEVEL_DATA
BuffAddr_StockLevel() { return
&m_txn.StockLevel; };
    inline PORDER_STATUS_DATA
BuffAddr_OrderStatus() { return
&m_txn.OrderStatus; };

    void NewOrder ( );
    void Payment ( );
    void Delivery ( );
    void StockLevel ( );
    void OrderStatus ( );

    // these are public because they
must be called from the dblib_err_handler and
msg_hangler
    // outside of the class
void SetDbLibError(int severity,
int dberr, int oserr, LPCSTR dberrstr, LPCSTR
oserrstr);
void SetSqlError( int msgno, int
msgstate, int severity, LPCSTR msgtext );
};

extern "C" DllDecl CTPCC_DBLIB* CTPCC_DBLIB_new
( LPCSTR szServer, LPCSTR szUser, LPCSTR
szPassword, LPCSTR szHost, LPCSTR szDatabase );

typedef CTPCC_DBLIB* (TYPE_CTPCC_DBLIB)(LPCSTR, LPCSTR,
LPCSTR, LPCSTR, LPCSTR);

tm_com_dll/src/tpcc_com.cpp
/* FILE: TPCC_COM.CPP Microsoft TPC-C
Kit Ver. 4.20.000 Copyright
Microsoft, 1999
* All Rights Reserved
* not yet audited
*
* PURPOSE: Source file for TPC-C COM+ class
implementation.
* Contact: Charles Levine
(clevine@microsoft.com)
*
* Change history:
* 4.20.000 - first version
*/

```

```

// needed for CoInitializeEx
#define _WIN32_WINNT 0x0400
#include <windows.h>

// need to declare functions for export
#define DllDecl __declspec( dllexport )

#include "..\..\common\src\trans.h"
//tpckit transaction header contains
definitions of structures specific to TPC-C
#include "..\..\common\src\error.h"
#include "..\..\common\src\txn_base.h"
#include "tpcc_com.h"

#include "..\..\tpcc_com_ps\src\tpcc_com_ps_i.c"
#include "..\..\tpcc_com_all\src\tpcc_com_all_i.c"

// wrapper routine for class constructor
__declspec(dllexport) CTPCC_COM* CTPCC_COM_new(BOOL
bSinglePool)
{
    return new CTPCC_COM(bSinglePool);
}

CTPCC_COM::CTPCC_COM(BOOL bSinglePool)
{
    HRESULT hr = NULL;
    long lRet = 0;
    ULONG ulTmpSize = 0;

    m_pTxn = NULL;
    m_pNewOrder = NULL;
    m_pPayment = NULL;
    m_pStockLevel = NULL;
    m_pOrderStatus = NULL;
    m_bSinglePool = bSinglePool;

    ulTmpSize = (ULONG) sizeof(COM_DATA);
    VariantInit(&m_vTxn);
    m_vTxn.vt = VT_SAFEARRAY;

    m_vTxn.parray = SafeArrayCreateVector(VT_UI1,
ulTmpSize, ulTmpSize);
    if (!m_vTxn.parray)
        throw new CCOMERR( E_FAIL );

    memset((void*)m_vTxn.parray-
>pvData,0,ulTmpSize);
    m_pTxn = (COM_DATA*)m_vTxn.parray->pvData;

    hr = CoInitializeEx(NULL,
COINIT_MULTITHREADED);
    if (FAILED(hr))
    {
        throw new CCOMERR( hr );
    }

    // create components
    if (m_bSinglePool)
    {
        hr = CoCreateInstance(CLSID_TPCC,
NULL, CLSCTX_SERVER, IID_ITPCC, (void *)&m_pNewOrder);
        if (FAILED(hr))
            throw new CCOMERR(hr);

        // all txns will use same component
        m_pPayment = m_pNewOrder;
        m_pStockLevel = m_pNewOrder;
        m_pOrderStatus = m_pNewOrder;
    }
}

```

```

else
{
    // use different components for
each txn
    hr =
CoCreateInstance(CLSID_NewOrder, NULL, CLSCTX_SERVER,
IID_ITPCC, (void **)&m_pNewOrder);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr =
CoCreateInstance(CLSID_Payment, NULL, CLSCTX_SERVER,
IID_ITPCC, (void **)&m_pPayment);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr =
CoCreateInstance(CLSID_StockLevel, NULL, CLSCTX_SERVER,
IID_ITPCC, (void **)&m_pStockLevel);
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr =
CoCreateInstance(CLSID_OrderStatus, NULL,
CLSCTX_SERVER, IID_ITPCC, (void **)&m_pOrderStatus);
    if (FAILED(hr))
        throw new CCOMERR(hr);
}
// call setcomplete to release each component
back into pool
hr = m_pNewOrder->CallSetComplete();
if (FAILED(hr))
    throw new CCOMERR(hr);

if (!m_bSinglePool)
{
    hr = m_pPayment->CallSetComplete();
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = m_pStockLevel->
CallSetComplete();
    if (FAILED(hr))
        throw new CCOMERR(hr);

    hr = m_pOrderStatus->
CallSetComplete();
    if (FAILED(hr))
        throw new CCOMERR(hr);
}
}

CTPCC_COM::~CTPCC_COM()
{
    if (m_pTxn)
        SafeArrayDestroy(m_pTxn.parray);

    ReleaseInterface(m_pNewOrder);
    if (!m_bSinglePool)
    {
        ReleaseInterface(m_pPayment);
        ReleaseInterface(m_pStockLevel);
        ReleaseInterface(m_pOrderStatus);
    }
    Counitialize();
}

void CTPCC_COM::NewOrder()
{
    VARIANT vTxn_out;

```

```

        HRESULT hr = m_pNewOrder->NewOrder(m_pTxn,
&vTxn_out);
        if (FAILED(hr))
            throw new CCOMERR( hr );
        memcpy(m_pTxn, (void *)vTxn_out.parray-
>pvData, vTxn_out.parray->rgsabound[0].cElements);
        SafeArrayDestroy(vTxn_out.parray);

        if ( m_pTxn->ErrorType != ERR_SUCCESS )
            throw new CCOMERR( m_pTxn-
>ErrorType, m_pTxn->error );
}

void CTPCC_COM::Payment()
{
    VARIANT vTxn_out;
    HRESULT hr = m_pPayment->Payment(m_pTxn,
&vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void *)vTxn_out.parray-
>pvData, vTxn_out.parray->rgsabound[0].cElements);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn-
>ErrorType, m_pTxn->error );
}

void CTPCC_COM::StockLevel()
{
    VARIANT vTxn_out;
    HRESULT hr = m_pStockLevel-
>StockLevel(m_pTxn, &vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void *)vTxn_out.parray-
>pvData, vTxn_out.parray->rgsabound[0].cElements);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn-
>ErrorType, m_pTxn->error );
}

void CTPCC_COM::OrderStatus()
{
    VARIANT vTxn_out;
    HRESULT hr = m_pOrderStatus-
>OrderStatus(m_pTxn, &vTxn_out);
    if (FAILED(hr))
        throw new CCOMERR( hr );
    memcpy(m_pTxn, (void *)vTxn_out.parray-
>pvData, vTxn_out.parray->rgsabound[0].cElements);
    SafeArrayDestroy(vTxn_out.parray);

    if ( m_pTxn->ErrorType != ERR_SUCCESS )
        throw new CCOMERR( m_pTxn-
>ErrorType, m_pTxn->error );
}

```

tm_com_dll/src/tpcc_com.h

```

/* FILE: TPCC_COM.H
 * Microsoft TPC-C
 * Kit Ver. 4.20.000
 * Copyright
 * Microsoft, 1999
 * All Rights Reserved
 * not yet audited
 *
 * PURPOSE: Header file for TPC-C COM+ class
 * implementation.
 * Change history:
 * 4.20.000 - first version
 */

#pragma once

#include <stdio.h>
#include "..\..\tpcc_com_ps\src\tpcc_com_ps.h"

// need to declare functions for import, unless define
has already been created
// by the DLL's .cpp module for export.
#ifdef DllDecl
#define DllDecl __declspec( dllimport )
#endif

class CCOMERR : public CBaseErr
{
private:
    char m_szErrorText[64];

public:
    // use this interface for genuine
COM errors
    CCOMERR( HRESULT hr )
    {
        m_hr = hr;
        m_iErrorType = 0;
        m_iError = 0;
    }

    // use this interface to
impersonate a non-COM error type
    CCOMERR( int iErrorType, int iError
)
    {
        m_iErrorType =
iErrorType;
        m_iError = iError;
        m_hr = S_OK;
    }

    int m_hr;
    int m_iErrorType;
    int m_iError;

    // A CCOMERR class can impersonate
another class, which happens if the error
// was not actually a COM Services
error, but was simply transmitted back via COM.
    int ErrorType()
    {
        if (m_iErrorType == 0)
            return
ERR_TYPE_COM;
        else
            return
m_iErrorType;
    }
}

```



```

int ErrorNum() {return m_hr;}
char *ErrorText()
{
    if (m_hr == S_OK)
        sprintf(
m_szErrorText, "Error: Class %d, error # %d",
m_iErrorType, m_iError );
    else
        sprintf(
m_szErrorText, "Error: COM HRESULT %x", m_hr );
    return m_szErrorText;
};

class DllDecl CTPCC_COM : public CTPCC_BASE
{
private:
    BOOL m_bSinglePool;

    // COM Interface pointers
    ITPCC*
m_pNewOrder;
    ITPCC*
m_pPayment;
    ITPCC*
m_pStockLevel;
    ITPCC*
m_pOrderStatus;

    struct COM_DATA
    {
        int ErrorType;
        int error;
        union
        {
            NEW_ORDER_DATA
            Payment;
            DELIVERY_DATA
            StockLevel;
            orderStatus;
        } u;
    };
    *m_pTxn;

public:
    VARIANT m_vTxn;
    CTPCC_COM(BOOL bSinglePool);
    ~CTPCC_COM(void);

    inline PNEW_ORDER_DATA
    BuffAddr_NewOrder() { return
&m_pTxn->u.NewOrder; };
    inline PPAYMENT_DATA
    BuffAddr_Payment() { return
&m_pTxn->u.Payment; };
    inline PDELIVERY_DATA
    BuffAddr_Delivery() { return
&m_pTxn->u.Delivery; };
    inline PSTOCK_LEVEL_DATA
    BuffAddr_StockLevel() { return
&m_pTxn->u.StockLevel; };
    inline PORDER_STATUS_DATA
    BuffAddr_OrderStatus() { return
&m_pTxn->u.OrderStatus; };

    void NewOrder        ();
    void Payment         ();
};

```

```

void StockLevel      ();
void OrderStatus    ();
void Delivery        ();
throw new CCOMERR(E_NOTIMPL); } // not supported
};

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

// wrapper routine for class constructor
extern "C" __declspec(dllexport) CTPCC_COM*
CTPCC_COM_new(BOOL);

typedef CTPCC_COM* (TYPE_CTPCC_COM)(BOOL);

tpcc_com_all/src/methods.h
/* FILE: METHODS.H Microsoft TPC-C
Kit Ver. 4.20.000 Copyright
Microsoft, 1999
All Rights Reserved
not yet audited
PURPOSE: Header file for COM components.
Change history: 4.20.000 - first version
*/

enum COMPONENT_ERROR
{
    ERR_MISSING_REGISTRY_ENTRIES = 1,
    ERR_LOADDLL_FAILED,
    ERR_GETPROCADDR_FAILED,
    ERR_UNKNOWN_DB_PROTOCOL
};

class CCOMPONENT_ERR : public CBaseErr
{
public:
    CCOMPONENT_ERR(COMPONENT_ERROR Err)
    {
        m_Error = Err;
        m_szTextDetail = NULL;
        m_SystemErr = 0;
        m_szErrorText = NULL;
    };

    CCOMPONENT_ERR(COMPONENT_ERROR Err,
    DWORD dwSystemErr)
    {
        m_Error = Err;
        m_szTextDetail = new
char[strlen(szTextDetail)+1];
        strcpy( m_szTextDetail,
szTextDetail );
        m_SystemErr =
dwSystemErr;
        m_szErrorText = NULL;
    };
};

```

```

};
~CCOMPONENT_ERR()
{
    if (m_szTextDetail !=
NULL)
        delete []
m_szTextDetail;
    if (m_szErrorText !=
NULL)
        delete []
m_szErrorText;
};

COMPONENT_ERROR m_Error;
char
*m_szTextDetail;
char
*m_szErrorText;
DWORD
m_SystemErr;

int ErrorType() {return
ERR_TYPE_COMPONENT;};
int ErrorNum() {return m_Error;};
char *ErrorText();

static void writeMessageToEventLog(LPTSTR lpszMsg);

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

    CTPCC_Common();
    ~CTPCC_Common();

// ITPCC
public:
    HRESULT __stdcall NewOrder(
    VARIANT txn_in, VARIANT* txn_out);
    HRESULT __stdcall Payment(
    VARIANT txn_in, VARIANT* txn_out);
    HRESULT __stdcall Delivery(
    VARIANT txn_in, VARIANT* txn_out) {return
E_NOTIMPL;};
    HRESULT __stdcall StockLevel( VARIANT txn_in,
    VARIANT* txn_out);
    HRESULT __stdcall OrderStatus(
    VARIANT txn_in, VARIANT* txn_out);

    HRESULT __stdcall CallSetComplete();

// IObjectContext
    STDMETHODIMP_(BOOL) CanBePooled() { return
m_bCanBePooled; }
};

```



```

tpcc_com_all/src/tpcc_com_all.cpp
/*      FILE:          TPCC_COM_ALL.CPP
 *      Microsoft TPC-C
Kit Ver. 4.20.000
 *      Copyright
Microsoft, 1999
 *      All Rights Reserved
 *
 *      Version
4.10.000 audited by Richard Gimarc, Performance
Metrics, 3/17/99
 *
 *      PURPOSE: Implementation for TPC-C Tuxedo
class.
 *      Contact: Charles Levine
(clevine@microsoft.com)
 *
 *      Change history: 4.20.000 - updated rev number to
match kit
 */

#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 <atimpl.cpp>
#include <comsvcs.h>

#include <sqltypes.h>
#include <sql.h>
#include <sqlext.h>

#include "tpcc_com_ps.h"
#include "..\..\common\src\trans.h"
//tpckit transaction
header contains definations of structures specific to
TPC-C
#include "..\..\common\src\txn_base.h"
#include "..\..\common\src\error.h"
#include "..\..\common\src\ReadRegistry.h"
#include "..\..\db_dblib_dll\src\tpcc_dblib.h"
// DBLIB implementation of TPC-C txns
#include "..\..\db_odbc_dll\src\tpcc_odbc.h"
// ODBC implementation of TPC-C txns

#include "resource.h"
#include "tpcc_com_all.h"
#include "tpcc_com_all_i.c"
#include "Methods.h"
#include "..\..\tpcc_com_ps\src\tpcc_com_ps_i.c"
#include "..\..\common\src\ReadRegistry.cpp"

CComModule _Module;

BEGIN_OBJECT_MAP(ObjectMap)
OBJECT_ENTRY(CLSID_TPCC, CTPCC)
OBJECT_ENTRY(CLSID_NewOrder, CNewOrder)
OBJECT_ENTRY(CLSID_OrderStatus, COrderStatus)
OBJECT_ENTRY(CLSID_Payment, CPayment)

```

```

OBJECT_ENTRY(CLSID_StockLevel, CStockLevel)
END_OBJECT_MAP()

// configuration settings from registry
TPCCREGISTRYDATA Reg;
char
szMyComputerName[MAX_COMPUTERNAME_LENGTH+1];

static HINSTANCE hLibInstanceDb = NULL;

TYPE_CTPCC_DBLIB *pCTPCC_DBLIB_new;
TYPE_CTPCC_ODBC *pCTPCC_ODBC_new;

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

extern "C"
BOOL WINAPI DllMain(HINSTANCE hInstance, DWORD
dwReason, LPVOID /*lpReserved*/)
{
char szDllName[128];

try
{
if (dwReason == DLL_PROCESS_ATTACH)
{
_Module.Init(ObjectMap,
hInstance);

DisableThreadLibraryCalls(hInstance);

DWORD dwSize =
MAX_COMPUTERNAME_LENGTH+1;

GetComputerName(szMyComputerName, &dwSize);
szMyComputerName[dwSize]
= 0;

if (
ReadTPCCRegistrySettings( &Reg ) )
throw new
CCOMPONENT_ERR( ERR_MISSING_REGISTRY_ENTRIES );

if (Reg.eDB_Protocol ==
DBLIB)
{
strcpy(
szDllName, Reg.szPath );
strcat(
szDllName, "tpcc_dblib.dll");

hLibInstanceDb
= LoadLibrary( szDllName );

if
(hLibInstanceDb == NULL)
throw
new CCOMPONENT_ERR( ERR_LOADDLL_FAILED, szDllName,
GetLastError() );

// get function
pointer to wrapper for class constructor
pCTPCC_DBLIB_new = (TYPE_CTPCC_DBLIB*)
GetProcAddress(hLibInstanceDb, "CTPCC_DBLIB_new");
if
(pCTPCC_DBLIB_new == NULL)
throw
new CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED, szDllName,
GetLastError() );
}
}
}
}

```

```

else if (Reg.eDB_Protocol
== ODBC)
{
strcpy(
szDllName, Reg.szPath );
strcat(
szDllName, "tpcc_odbc.dll");

hLibInstanceDb
= LoadLibrary( szDllName );

if
(hLibInstanceDb == NULL)
throw
new CCOMPONENT_ERR( ERR_LOADDLL_FAILED, szDllName,
GetLastError() );

// get function
pointer to wrapper for class constructor
pCTPCC_ODBC_new
= (TYPE_CTPCC_ODBC*)
GetProcAddress(hLibInstanceDb, "CTPCC_ODBC_new");
if
(pCTPCC_ODBC_new == NULL)
throw
new CCOMPONENT_ERR( ERR_GETPROCADDR_FAILED, szDllName,
GetLastError() );
}
else
throw new
CCOMPONENT_ERR( ERR_UNKNOWN_DB_PROTOCOL );
}
else if (dwReason ==
DLL_PROCESS_DETACH)
{
_Module.Term();
}
}
catch (CBaseErr *e)
{
WriteMessageToEventLog(e-
>ErrorText());
delete e;
return FALSE;
}
catch (...)
{
WriteMessageToEventLog(TEXT("Unhandled
exception in object DllMain"));
return FALSE;
}
}
return TRUE; // OK

////////////////////////////////////
// Used to determine whether the DLL can be unloaded by
OLE
STDAPI DllCanUnloadNow(void)
{
return (_Module.GetLockCount()==0) ? S_OK :
S_FALSE;
}

////////////////////////////////////
// Returns a class factory to create an object of the
requested type
STDAPI DllGetObject(REFCLSID rclsid, REFIID riid,
LPVOID* ppv)
{

```

```

        return _Module.GetClassObject(rcClsid, riid,
    ppv);
}

////////////////////////////////////
// DLLRegisterServer - Adds entries to the system
// registry
STDAPI DLLRegisterServer(void)
{
    // registers object, typelib and all
    // interfaces in typelib
    return _Module.RegisterServer(TRUE);
}

////////////////////////////////////
// DLLUnregisterServer - Removes entries from the
// system registry
STDAPI DLLUnregisterServer(void)
{
    _Module.UnregisterServer();
    return S_OK;
}

static void writeMessageToEventLog(LPTSTR lpszMsg)
{
    TCHAR szMsg[256];
    HANDLE hEventSource;
    LPTSTR lpszStrings[2];

    // Use event logging to log the error.
    hEventSource = RegisterEventSource(NULL,
    TEXT("tpcc_com_all.dll"));

    _stprintf(szMsg, TEXT("Error in COM+ TPC-C
    Component: "));
    lpszStrings[0] = szMsg;
    lpszStrings[1] = lpszMsg;

    if (hEventSource != NULL)
    {
        ReportEvent(hEventSource, // handle of event
        source
            EVENTLOG_ERROR_TYPE, // event type
            0, // event category
            0, // event ID
            NULL, // current user's SID
            2, // strings in
        lpszStrings
            0, // no bytes of raw
        data
            (LPCTSTR *)lpszStrings, // array of
        error strings
            NULL); // no raw data

        (VOID) DeregisterEventSource(hEventSource);
    }
}

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

```

```

/* FUNCTION: CCOMPONENT_ERR::ErrorText
*/
char* CCOMPONENT_ERR::ErrorText(void)
{
    static SERRORMSG errorMsgs[] =
    {
        { ERR_MISSING_REGISTRY_ENTRIES,
        "Required entries missing from registry." },
        { ERR_LOADDLL_FAILED,
        "Load of DLL failed. DLL="
        },
        { ERR_GETPROCADDR_FAILED,
        "Could not map proc in DLL. GetProcAddress
        error. DLL="
        },
        { ERR_UNKNOWN_DB_PROTOCOL,
        "Unknown database protocol specified in
        registry."
        },
        { 0, ""
        }
    };

    char szTmp[256];
    int i = 0;
    while (TRUE)
    {
        if (errorMsgs[i].szMsg[0] == 0)
        {
            strcpy( szTmp, "Unknown
            error number." );
            break;
        }
        if (m_Error == errorMsgs[i].iError)
        {
            strcpy( szTmp,
            errorMsgs[i].szMsg );
            break;
        }
        i++;
    }

    if (m_szTextDetail)
        strcat( szTmp, m_szTextDetail );
    if (m_SystemErr)
        wsprintf( szTmp+strlen(szTmp), "
        Error=%d", m_SystemErr );

    m_szErrorText = new char[strlen(szTmp)+1];
    strcpy( m_szErrorText, szTmp );
    return m_szErrorText;
}

CTPCC_Common::CTPCC_Common()
{
    m_pTxn = NULL;
    m_bCanBePooled = TRUE;
}

CTPCC_Common::~~CTPCC_Common()
{
    if (m_pTxn)
        delete m_pTxn;
}

```

```

HRESULT CTPCC_Common::CallSetComplete()
{
    IObjectContext* pObjContext = NULL;

    // get our object context
    HRESULT hr = CoGetObjectContext(
    IID_IObjectContext, (void **)&pObjContext );
    pObjContext->SetComplete();
    ReleaseInterface(pObjContext);
    return hr;
}

// called by the ctor activator
STDMETHODIMP CTPCC_Common::Construct(IDispatch * punk)
{
    // Code to access construction string, if
    // needed later...
    if (!punk)
        return E_UNEXPECTED;
    IObjectConstructString * pString =
    NULL;
    HRESULT hr = punk-
    >QueryInterface(IID_IObjectConstructString, (void
    **)&pString);
    pString->Release();

    try
    {
        if (Reg.eDB_Protocol == ODBC)
            m_pTxn = pCTPCC_ODBC_new(
            Reg.szDbServer, Reg.szDbUser, Reg.szDbPassword,
            Reg.szMyComputerName, Reg.szSPPrefix );
        else if (Reg.eDB_Protocol == DBLIB)
            m_pTxn =
            pCTPCC_DBLIB_new( Reg.szDbServer, Reg.szDbUser,
            Reg.szDbPassword, szMyComputerName, Reg.szDbName );
    }
    catch (CBaseErr *e)
    {
        WriteMessageToEventLog(e-
        >ErrorText());
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
        exception in object ::Construct"));
        return E_FAIL;
    }

    return S_OK;
}

HRESULT CTPCC_Common::NewOrder(VARIANT txn_in, VARIANT*
    txn_out)
{
    PNEW_ORDER_DATA pNewOrder;
    COM_DATA *pData;
    try
    {
        pData = (COM_DATA*)txn_in.parray-
        >pvData;
        pNewOrder = m_pTxn-
        >BuffAddr_NewOrder();
        memcpy(pNewOrder, &pData-
        >u.NewOrder, sizeof(NEW_ORDER_DATA));
    }
}

```



```

        return S_OK;
    }
    catch (CBaseErr *e)
    {
        // check for lost database
        // connection; if yes, component is toast
        if ( ((e->ErrorType() ==
ERR_TYPE_DBLIB) && (e->ErrorNum() == 10005)) ||
            ((e->ErrorType() ==
ERR_TYPE_ODBC) && (e->ErrorNum() == 10054)) )
            m_bCanBePooled = FALSE;

        pData->retval = e->ErrorType();
        pData->error = e->ErrorNum();
        delete e;
        return E_FAIL;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled
exception."));
        pData->retval = ERR_TYPE_LOGIC;
        pData->error = 0;
        m_bCanBePooled = FALSE;
        return E_FAIL;
    }
}
}

```

tpcc_com_all/src/tpcc_com_all.def

; tpcc_com_all.def : Declares the module parameters.

```

LIBRARY      "tpcc_com_all.dll"

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

```

tpcc_com_all/src/tpcc_com_all.h

```

#pragma warning( disable: 4049 ) /* more than 64k
source lines */

/* this ALWAYS GENERATED file contains the definitions
for the interfaces */

/* File created by MIDL compiler version 6.00.0347 */
/* at Fri Aug 01 10:56:27 2003
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
  Oicf, w1, Zp8, env=win32 (32b run)
  protocol : dce , ms_ext, c_ext
  error checks: allocation ref bounds_check enum
stub_data
  VC __declspec( decoration level:
  __declspec(uuid()), __declspec(selectany),
  __declspec(novtable)
  DECLSPEC_UUID(), MIDL_INTERFACE()
  */
//@@MIDL_FILE_HEADING( )

```

```

/* verify that the <rpcndr.h> version is high enough to
compile this file*/
#ifdef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

#include "rpc.h"
#include "rpcndr.h"

#ifdef __tpcc_com_all_h__
#define __tpcc_com_all_h__

#ifdef _MSC_VER && (_MSC_VER >= 1020)
#pragma once
#endif

/* Forward Declarations */

#ifdef __TPCC_FWD_DEFINED__
#define __TPCC_FWD_DEFINED__

#ifdef __cplusplus
typedef class TPCC TPCC;
#else
typedef struct TPCC TPCC;
#endif /* __cplusplus */
#endif /* __TPCC_FWD_DEFINED__ */

#ifdef __NewOrder_FWD_DEFINED__
#define __NewOrder_FWD_DEFINED__

#ifdef __cplusplus
typedef class NewOrder NewOrder;
#else
typedef struct NewOrder NewOrder;
#endif /* __cplusplus */
#endif /* __NewOrder_FWD_DEFINED__ */

#ifdef __OrderStatus_FWD_DEFINED__
#define __OrderStatus_FWD_DEFINED__

#ifdef __cplusplus
typedef class OrderStatus OrderStatus;
#else
typedef struct OrderStatus OrderStatus;
#endif /* __cplusplus */
#endif /* __OrderStatus_FWD_DEFINED__ */

#ifdef __Payment_FWD_DEFINED__
#define __Payment_FWD_DEFINED__

#ifdef __cplusplus
typedef class Payment Payment;
#else
typedef struct Payment Payment;
#endif /* __cplusplus */
#endif /* __Payment_FWD_DEFINED__ */

#ifdef __StockLevel_FWD_DEFINED__
#define __StockLevel_FWD_DEFINED__

#ifdef __cplusplus
typedef class StockLevel StockLevel;

```

```

#else
typedef struct StockLevel StockLevel;
#endif /* __cplusplus */
#endif /* __StockLevel_FWD_DEFINED__ */

/* header files for imported files */
#include "oaidl.h"
#include "ocidl.h"
#include "tpcc_com_ps.h"

#ifdef __cplusplus
extern "C"{
#endif

void * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void * );

/* interface __MIDL_itf_tpcc_com_all_0000 */
/* [local] */

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_all_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_all_0000_v0_0_s_ifspec;

#ifdef __TPCClib_LIBRARY_DEFINED__
#define __TPCClib_LIBRARY_DEFINED__

/* library TPCClib */
/* [helpstring][version][uuid] */

EXTERN_C const IID LIBID_TPCClib;

EXTERN_C const CLSID CLSID_TPCC;

#ifdef __cplusplus
class DECLSPEC_UUID("122A3128-2520-11D3-BA71-00C04FBFE08B")
TPCC;
#endif

EXTERN_C const CLSID CLSID_NewOrder;

#ifdef __cplusplus
class DECLSPEC_UUID("975BAABF-84A7-11D2-BA47-00C04FBFE08B")
NewOrder;
#endif

EXTERN_C const CLSID CLSID_OrderStatus;

#ifdef __cplusplus
class DECLSPEC_UUID("266836AD-A50D-11D2-BA4E-00C04FBFE08B")
OrderStatus;
#endif

EXTERN_C const CLSID CLSID_Payment;

```

```

#ifdef __cplusplus
class DECLSPEC_UUID("CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B")
Payment;
#endif

EXTERN_C const CLSID CLSID_StockLevel;

#ifdef __cplusplus
class DECLSPEC_UUID("2668369E-A50D-11D2-BA4E-00C04FBFE08B")
StockLevel;
#endif /* __TPCCLib_LIBRARY_DEFINED__ */
/* Additional Prototypes for ALL interfaces */
/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif
#endif

```

tpcc_com_all/src/tpcc_com_all.idl

```

/* FILE: TPCC.IDL Microsoft TPC-C
Kit Ver. 4.20.000 Copyright
Microsoft, 1999 All Rights Reserved
* not yet audited
*
* PURPOSE: IDL source for TPCC.dll. This file
is processed by the MIDL tool to
produce the
type library (TPCC.tlb) and marshalling code.
* Change history:
* 4.20.000 - first version
*/

interface TPCC;
interface NewOrder;
interface OrderStatus;
interface Payment;
interface StockLevel;

```

```

import "oidl.idl";
import "ocidl.idl";
import "..\tpcc_com_ps\src\tpcc_com_ps.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;
};

[
    uuid(975BAABF-84A7-11D2-BA47-00C04FBFE08B),
    helpstring("NewOrder Class")
]
coclass NewOrder
{
    [default] interface ITPCC;
};

[
    uuid(266836AD-A50D-11D2-BA4E-00C04FBFE08B),
    helpstring("OrderStatus Class")
]
coclass OrderStatus
{
    [default] interface ITPCC;
};

[
    uuid(CD02F7EF-A4FA-11D2-BA4E-00C04FBFE08B),
    helpstring("Payment Class")
]
coclass Payment
{
    [default] interface ITPCC;
};

[
    uuid(2668369E-A50D-11D2-BA4E-00C04FBFE08B),
    helpstring("StockLevel Class")
]
coclass StockLevel
{
    [default] interface ITPCC;
};
};

```

tpcc_com_all/src/tpcc_com_all.rc

```

//Microsoft Developer Studio generated resource script.
//

```

```

#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
////////////////////////////////////
// Generated from the TEXTINCLUDE 2 resource.
//
#include "winres.h"
////////////////////////////////////
#undef APSTUDIO_READONLY_SYMBOLS
// English (U.S.) resources

#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
#ifdef _WIN32
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US
#pragma code_page(1252)
#endif // _WIN32

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

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

3 TEXTINCLUDE DISCARDABLE
BEGIN
    "1 TYPELIB ""tpcc_com_all.tlb""\r\n"
    "\0"
END

#ifdef APSTUDIO_INVOKED

#ifdef _MAC
////////////////////////////////////
// Version
//

VS_VERSION_INFO VERSIONINFO
FILEVERSION 1,0,0,1
PRODUCTVERSION 1,0,0,1
FILEFLAGSMASK 0x3fL
#ifdef _DEBUG
FILEFLAGS 0x1L
#else
FILEFLAGS 0x0L
#endif
FILEOS 0x4L
FILETYPE 0x2L
FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"

```

```

BEGIN
  BLOCK "040904B0"
  BEGIN
    VALUE "CompanyName", "\0"
    VALUE "FileDescription", "tpcc_com_all
Module\0"
    VALUE "FileVersion", "1, 0, 0, 1\0"
    VALUE "InternalName", "TPCCNEWORDER\0"
    VALUE "LegalCopyright", "Copyright 1997\0"
    VALUE "OriginalFilename",
"tpcc_com_all.DLL\0"
    VALUE "ProductName", "tpcc_com_all
Module\0"
    VALUE "ProductVersion", "1, 0, 0, 1\0"
    VALUE "OLESelfRegister", "\0"
  END
  END
  BLOCK "VarFileInfo"
  BEGIN
    VALUE "Translation", 0x409, 1200
  END
  END
#endif // !_MAC

////////////////////////////////////
//
// REGISTRY
//
IDR_TPCC          REGISTRY DISCARDABLE
"tpcc_com_all.rgs"
IDR_NEWORDER     REGISTRY DISCARDABLE
"tpcc_com_no.rgs"
IDR_ORDERSTATUS REGISTRY DISCARDABLE
"tpcc_com_os.rgs"
IDR_PAYMENT      REGISTRY DISCARDABLE
"tpcc_com_pay.rgs"
IDR_STOCKLEVEL  REGISTRY DISCARDABLE
"tpcc_com_sl.rgs"

////////////////////////////////////
//
// String Table
//
STRINGTABLE DISCARDABLE
BEGIN
  IDS_PROJNAME      "tpcc_com_all"
END

#endif // English (U.S.) resources
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//
1 TYPELIB "tpcc_com_all.tlb"

////////////////////////////////////
//
// not APSTUDIO_INVOKED
#endif

```

tpcc_com_all/src/tpcc_com_all.rgs

```

HKCR
{
  TPCC.AllTxns.1 = s 'All Txns Class'
  {
    CLSID = s '{122A3128-2520-11D3-
BA71-00C04FBFE08B}'
    TPCC.AllTxns = s 'TPCC Class'
    {
      CurVer = s 'TPCC.AllTxns.1'
    }
    NoRemove CLSID
    {
      ForceRemove {122A3128-2520-11D3-
BA71-00C04FBFE08B} = s 'TPCC Class'
    }
    ProgID = s
    VersionIndependentProgID
    InprocServer32 = s
    '%MODULE%'
    {
      val
    }
  }
  ThreadingModel = s 'Both'
}

```

tpcc_com_all/src/tpcc_com_all.i.c

```

#pragma warning( disable: 4049 ) /* more than 64k
source lines */

/* this ALWAYS GENERATED file contains the IIDs and
CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 6.00.0347 */
/* at Fri Aug 01 10:56:27 2003 */
/* Compiler settings for .\src\tpcc_com_all.idl:
  oicf, w1, Zp8, env=win32 (32b run)
  protocol : dce , ms_ext, c_ext
  error checks: allocation ref bounds_check enum
  stub_data
  VC __declspec( decoration level:
  __declspec(uuid()), __declspec(selectany),
  __declspec(novtable)
  DECLSPEC_UUID(), MIDL_INTERFACE()
  */
//@@MIDL_FILE_HEADING( )

#if !defined(_M_IA64) && !defined(_M_AMD64)

#ifdef __cplusplus
extern "C" {
#endif

```

```

#include <rpc.h>
#include <rpcndr.h>

#ifdef _MIDL_USE_GUIDDEF_

#ifndef INITGUID
#define INITGUID
#include <guiddef.h>
#undef INITGUID
#else
#include <guiddef.h>
#endif

#define MIDL_DEFINE_GUID(type, name, l, w1, w2, b1, b2, b3, b4, b5, b6, b7, b8) \
DEFINE_GUID(name, l, w1, w2, b1, b2, b3, b4, b5, b6, b7, b8)

#else // !_MIDL_USE_GUIDDEF_

#ifndef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
  unsigned long x;
  unsigned short s1;
  unsigned short s2;
  unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

#define MIDL_DEFINE_GUID(type, name, l, w1, w2, b1, b2, b3, b4, b5, b6, b7, b8) \
const type name = \
{l, w1, w2, {b1, b2, b3, b4, b5, b6, b7, b8}}

#endif !_MIDL_USE_GUIDDEF_

MIDL_DEFINE_GUID(IID,
LIBID_TPCCLib, 0x122A3128, 0x2520, 0x11D3, 0xBA, 0x71, 0x00, 0x00, 0x4F, 0xBF, 0xE0, 0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_TPCC, 0x122A3128, 0x2520, 0x11D3, 0xBA, 0x71, 0x00, 0x00, 0x4F, 0xBF, 0xE0, 0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_NewOrder, 0x975BAABF, 0x84A7, 0x11D2, 0xBA, 0x47, 0x00, 0x00, 0x4F, 0xBF, 0xE0, 0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_OrderStatus, 0x266836AD, 0xA50D, 0x11D2, 0xBA, 0x4E, 0x00, 0x00, 0x4F, 0xBF, 0xE0, 0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_Payment, 0xcd02f7ef, 0xa4fa, 0x11d2, 0xba, 0x4e, 0x00, 0x00, 0x4f, 0xbf, 0xe0, 0x8b);

```



```

MIDL_DEFINE_GUID(CLSID,
CLSID_StockLevel,0x2668369E,0xA50D,0x11D2,0xBA,0x4E,0x0,0x0,0x4F,0xBF,0xE0,0x8B);

#undef MIDL_DEFINE_GUID
#ifdef __cplusplus
}
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AMD64)*/

```

tpcc_com_all/src/tpcc_com_no.rgs

```

HKCR
{
    TPCC.NewOrder.1 = s 'NewOrder Class'
    {
        CLSID = s '{975BAABF-84A7-11D2-
BA47-00C04FBFE08B}'
    }
    TPCC.NewOrder = s 'NewOrder Class'
    {
        CurVer = s 'TPCC.NewOrder.1'
    }
    NoRemove CLSID
    {
        ForceRemove {975BAABF-84A7-11D2-
BA47-00C04FBFE08B} = s 'NewOrder Class'
    }
    ProgID = s
    VersionIndependentProgID
= s 'TPCC.NewOrder'
    InprocServer32 = s
    '%MODULE%'
    {
        val
    }
    ThreadingModel = s 'Both'
}
}
}

```

tpcc_com_all/src/tpcc_com_os.rgs

```

HKCR
{
    TPCC.OrderStatus.1 = s 'OrderStatus Class'
    {
        CLSID = s '{266836AD-A50D-11D2-
BA4E-00C04FBFE08B}'
    }
    TPCC.OrderStatus = s 'OrderStatus Class'
    {
        CurVer = s 'TPCC.OrderStatus.1'
    }
    NoRemove CLSID
    {

```

```

        ForceRemove {266836AD-A50D-11D2-
BA4E-00C04FBFE08B} = s 'OrderStatus Class'
    }
    ProgID = s
    VersionIndependentProgID
= s 'TPCC.OrderStatus'
    InprocServer32 = s
    '%MODULE%'
    {
        val
    }
    ThreadingModel = s 'Both'
}
}
}

```

tpcc_com_all/src/tpcc_com_pay.rgs

```

HKCR
{
    TPCC.Payment.1 = s 'Payment Class'
    {
        CLSID = s '{CD02F7EF-A4FA-11D2-
BA4E-00C04FBFE08B}'
    }
    TPCC.Payment = s 'Payment Class'
    {
        CurVer = s 'TPCC.Payment.1'
    }
    NoRemove CLSID
    {
        ForceRemove {CD02F7EF-A4FA-11D2-
BA4E-00C04FBFE08B} = s 'Payment Class'
    }
    ProgID = s
    VersionIndependentProgID
= s 'TPCC.Payment'
    InprocServer32 = s
    '%MODULE%'
    {
        val
    }
    ThreadingModel = s 'Both'
}
}
}

```

tpcc_com_all/src/tpcc_com_ps.h

```

#pragma warning( disable: 4049 ) /* more than 64k
source lines */

/* this ALWAYS GENERATED file contains the definitions
for the interfaces */

/* File created by MIDL compiler version 6.00.0347 */
/* at Fri Aug 01 10:56:14 2003
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:

```

```

Oicf, w1, zp8, env=win32 (32b run)
protocol : dce , ms_ext, c_ext
error checks: allocation ref bounds_check enum
stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany),
__declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING( )

```

```

/* verify that the <rpcndr.h> version is high enough to
compile this file*/
#ifndef __REQUIRED_RPCNDR_H_VERSION__
#define __REQUIRED_RPCNDR_H_VERSION__ 440
#endif

```

```

#include "rpc.h"
#include "rpcndr.h"

```

```

#ifndef __RPCNDR_H_VERSION__
#error this stub requires an updated version of
<rpcndr.h>
#endif // __RPCNDR_H_VERSION__

```

```

#ifndef COM_NO_WINDOWS_H
#include "windows.h"
#include "ole2.h"
#endif /*COM_NO_WINDOWS_H*/

```

```

#ifndef __tpcc_com_ps_h
#define __tpcc_com_ps_h

```

```

#if defined(_MSC_VER) && (_MSC_VER >= 1020)
#pragma once
#endif

```

```

/* Forward Declarations */

```

```

#ifndef __ITPCC_FWD_DEFINED__
#define __ITPCC_FWD_DEFINED__
typedef interface ITPCC ITPCC;
#endif /* __ITPCC_FWD_DEFINED__ */

```

```

/* header files for imported files */
#include "oaidl.h"
#include "ocidl.h"

```

```

#ifdef __cplusplus
extern "C"{
#endif

```

```

void * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void * );

```

```

/* interface __MIDL_itf_tpcc_com_ps_0000 */
/* [local] */

```

```

extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_ps_0000_v0_0_c_ifspec;
extern RPC_IF_HANDLE
__MIDL_itf_tpcc_com_ps_0000_v0_0_s_ifspec;

```

```

#ifndef __ITPCC_INTERFACE_DEFINED__
#define __ITPCC_INTERFACE_DEFINED__

```

```

/* interface ITPCC */

```

```

/* [unique][helpstring][uuid][oleautomation][object] */
EXTERN_C const IID IID_ITPCC;
#if defined(__cplusplus) && !defined(CINTERFACE)
    MIDL_INTERFACE("FEEE6AA2-84B1-11d2-BA47-00C04FBFE088")
    ITPCC : public IUnknown
    {
    public:
        virtual HRESULT __stdcall NewOrder(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT __stdcall Payment(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT __stdcall Delivery(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT __stdcall StockLevel(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT __stdcall OrderStatus(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT __stdcall CallSetComplete(
            void) = 0;
    };
#else /* C style interface */
    typedef struct ITPCCVtbl
    {
        BEGIN_INTERFACE

        HRESULT ( STDMETHODCALLTYPE *QueryInterface )(
            ITPCC * This,
            /* [in] */ REFIID riid,
            /* [iid_is][out] */ void **ppvObject);

        ULONG ( STDMETHODCALLTYPE *AddRef )(
            ITPCC * This);

        ULONG ( STDMETHODCALLTYPE *Release )(
            ITPCC * This);

        HRESULT ( STDMETHODCALLTYPE *NewOrder )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *Payment )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *Delivery )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE *StockLevel )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);
    };
#endif

```

```

/* [out] */ VARIANT *txn_out);
HRESULT ( STDMETHODCALLTYPE *OrderStatus )(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

HRESULT ( STDMETHODCALLTYPE *CallSetComplete )(
    ITPCC * This);

    END_INTERFACE
} ITPCCVtbl;

interface ITPCC
{
    CONST_VTBL struct ITPCCVtbl *lpVtbl;
};

#ifndef COBJMACROS

#define ITPCC_QueryInterface(This,riid,ppvObject) \
    (This->lpVtbl->QueryInterface(This,riid,ppvObject)
#define ITPCC_AddRef(This) \
    (This->lpVtbl->AddRef(This)
#define ITPCC_Release(This) \
    (This->lpVtbl->Release(This)

#define ITPCC_NewOrder(This,txn_in,txn_out) \
    (This->lpVtbl->NewOrder(This,txn_in,txn_out)
#define ITPCC_Payment(This,txn_in,txn_out) \
    (This->lpVtbl->Payment(This,txn_in,txn_out)
#define ITPCC_Delivery(This,txn_in,txn_out) \
    (This->lpVtbl->Delivery(This,txn_in,txn_out)
#define ITPCC_StockLevel(This,txn_in,txn_out) \
    (This->lpVtbl->StockLevel(This,txn_in,txn_out)
#define ITPCC_OrderStatus(This,txn_in,txn_out) \
    (This->lpVtbl->OrderStatus(This,txn_in,txn_out)
#define ITPCC_CallSetComplete(This) \
    (This->lpVtbl->CallSetComplete(This)

#endif /* COBJMACROS */

#endif /* C style interface */

HRESULT STDMETHODCALLTYPE ITPCC_NewOrder_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

```

```

HRESULT STDMETHODCALLTYPE ITPCC_Payment_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_Delivery_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_StockLevel_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_OrderStatus_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_CallSetComplete_Proxy(
    ITPCC * This);

void __RPC_STUB ITPCC_CallSetComplete_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

#endif /* __ITPCC_INTERFACE_DEFINED__ */

/* Additional Prototypes for ALL interfaces */
unsigned long __RPC_USER VARIANT_UserSize(
    unsigned long *, unsigned long, VARIANT *
);

```



```

#if defined(__cplusplus) && !defined(CINTERFACE)
    MIDL_INTERFACE("FE6E6AA2-84B1-11d2-BA47-00C04FBFE08B")
    ITPCC : public IUnknown
    {
    public:
        virtual HRESULT STDMETHODCALLTYPE NewOrder(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT STDMETHODCALLTYPE Payment(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT STDMETHODCALLTYPE Delivery(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT STDMETHODCALLTYPE StockLevel(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT STDMETHODCALLTYPE OrderStatus(
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out) = 0;

        virtual HRESULT STDMETHODCALLTYPE CallSetComplete(
            void) = 0;
    };

#else /* C style interface */
    typedef struct ITPCCVtbl
    {
        BEGIN_INTERFACE

        HRESULT ( STDMETHODCALLTYPE QueryInterface )(
            ITPCC * This,
            /* [in] */ REFIID riid,
            /* [iid_is][out] */ void **ppvObject);

        ULONG ( STDMETHODCALLTYPE AddRef )(
            ITPCC * This);

        ULONG ( STDMETHODCALLTYPE Release )(
            ITPCC * This);

        HRESULT ( STDMETHODCALLTYPE NewOrder )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE Payment )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE Delivery )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE StockLevel )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,
            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE OrderStatus )(
            ITPCC * This,
            /* [in] */ VARIANT txn_in,

```

```

            /* [out] */ VARIANT *txn_out);

        HRESULT ( STDMETHODCALLTYPE CallSetComplete )(
            ITPCC * This);

        END_INTERFACE
    } ITPCCVtbl;

    interface ITPCC
    {
        CONST_VTBL struct ITPCCVtbl *lpVtbl;
    };

#ifdef COBJMACROS

#define ITPCC_QueryInterface(This,riid,ppvObject) \
    (This->lpVtbl->QueryInterface(This,riid,ppvObject)

#define ITPCC_AddRef(This) \
    (This->lpVtbl->AddRef(This)

#define ITPCC_Release(This) \
    (This->lpVtbl->Release(This)

#define ITPCC_NewOrder(This,txn_in,txn_out) \
    (This->lpVtbl->NewOrder(This,txn_in,txn_out)

#define ITPCC_Payment(This,txn_in,txn_out) \
    (This->lpVtbl->Payment(This,txn_in,txn_out)

#define ITPCC_Delivery(This,txn_in,txn_out) \
    (This->lpVtbl->Delivery(This,txn_in,txn_out)

#define ITPCC_StockLevel(This,txn_in,txn_out) \
    (This->lpVtbl->StockLevel(This,txn_in,txn_out)

#define ITPCC_OrderStatus(This,txn_in,txn_out) \
    (This->lpVtbl->OrderStatus(This,txn_in,txn_out)

#define ITPCC_CallSetComplete(This) \
    (This->lpVtbl->CallSetComplete(This)

#endif /* COBJMACROS */

#endif /* C style interface */

HRESULT STDMETHODCALLTYPE ITPCC_NewOrder_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_Payment_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

```

```

void __RPC_STUB ITPCC_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_Delivery_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_StockLevel_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_OrderStatus_Proxy(
    ITPCC * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT *txn_out);

void __RPC_STUB ITPCC_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

HRESULT STDMETHODCALLTYPE ITPCC_CallSetComplete_Proxy(
    ITPCC * This);

void __RPC_STUB ITPCC_CallSetComplete_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *_pRpcChannelBuffer,
    PRPC_MESSAGE _pRpcMessage,
    DWORD *_pdwStubPhase);

#endif /* __ITPCC_INTERFACE_DEFINED__ */

/* Additional Prototypes for ALL interfaces */
unsigned long __RPC_USER VARIANT_UserSize(
    unsigned long *, unsigned long *, VARIANT *
);
unsigned char * __RPC_USER VARIANT_UserMarshal(
    unsigned long *, unsigned char *, VARIANT * );
unsigned char * __RPC_USER
VARIANT_UserUnmarshal(unsigned long *, unsigned char *,
    VARIANT * );

```

```

void
unsigned long *, VARIANT *);
/* end of Additional Prototypes */
#ifdef __cplusplus
}
#endif
#endif

tpcc_com_ps/src/tpcc_com_ps.idl
/* FILE: ITPCC.IDL Microsoft TPC-C
Kit Ver. 4.20.000 Copyright
Microsoft, 1999 All Rights Reserved
*
* not yet audited
*
* PURPOSE: Defines the interface used by TPCC.
This interface can be implemented by C++ components.
* Change history: 4.20.000 - first version
*/

// Forward declare all types defined
interface ITPCC;
import "oidl.idl";
import "ocidl.idl";

[
    object,
    oleautomation,
    uuid(FEEE6AA2-84B1-11d2-BA47-
00C04FBFE08B),
    helpstring("ITPCC Interface"),
    pointer_default(unique)
]
interface ITPCC : IUnknown
{
    HRESULT STDMETHODCALLTYPE NewOrder
    (
        [in] VARIANT txn_in,
        [out] VARIANT *txn_out
    );
    HRESULT STDMETHODCALLTYPE Payment
    (
        [in] VARIANT txn_in,
        [out] VARIANT *txn_out
    );
    HRESULT STDMETHODCALLTYPE Delivery
    (
        [in] VARIANT txn_in,

```

```

[out] VARIANT *txn_out
);
HRESULT STDMETHODCALLTYPE StockLevel
(
    [in] VARIANT txn_in,
    [out] VARIANT *txn_out
);
HRESULT STDMETHODCALLTYPE OrderStatus
(
    [in] VARIANT txn_in,
    [out] VARIANT *txn_out
);
HRESULT STDMETHODCALLTYPE CallSetComplete
(
);
}; // interface ITPCC

tpcc_com_ps/src/tpcc_com_ps.i.c

#pragma warning( disable: 4049 ) /* more than 64k
source lines */

/* this ALWAYS GENERATED file contains the IIDs and
CLSIDs */

/* link this file in with the server and any clients */

/* File created by MIDL compiler version 6.00.0347 */
/* at Fri Aug 01 10:56:14 2003
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf, w1, Zp8, env=win32 (32b run)
protocol: dce, ms_ext, c_ext
error checks: allocation ref bounds_check enum
stub_data
VC __declspec() decoration level:
__declspec(uuid()), __declspec(selectany),
__declspec(novtable)
DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@MIDL_FILE_HEADING( )

#if !defined(_M_IA64) && !defined(_M_AMD64)

#ifdef __cplusplus
extern "C" {
#endif

#include <rpc.h>
#include <rpcndr.h>

```

```

#ifdef _MIDL_USE_GUIDDEF_

#ifndef INITGUID
#define INITGUID
#include <guiddef.h>
#undef INITGUID
#else
#include <guiddef.h>
#endif

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
DEFINE_GUID(name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8)

#else // !_MIDL_USE_GUIDDEF_

#ifndef __IID_DEFINED__
#define __IID_DEFINED__

typedef struct _IID
{
    unsigned long x;
    unsigned short s1;
    unsigned short s2;
    unsigned char c[8];
} IID;

#endif // __IID_DEFINED__

#ifndef CLSID_DEFINED
#define CLSID_DEFINED
typedef IID CLSID;
#endif // CLSID_DEFINED

#define MIDL_DEFINE_GUID(type,name,l,w1,w2,b1,b2,b3,b4,b5,b6,b7,b8) \
const type name = \
{1,w1,w2,{b1,b2,b3,b4,b5,b6,b7,b8}}

#endif // !_MIDL_USE_GUIDDEF_

MIDL_DEFINE_GUID(IID,
IID_ITPCC,0xFEEE6AA2,0x84B1,0x11d2,0xBA,0x47,0x00,0xC0,
0x4F,0xBF,0xE0,0x8B);

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif // !defined(_M_IA64) && !defined(_M_AMD64)*/

tpcc_com_ps/src/tpcc_com_ps.p.c

#pragma warning( disable: 4049 ) /* more than 64k
source lines */

/* this ALWAYS GENERATED file contains the proxy stub
code */

/* File created by MIDL compiler version 6.00.0347 */
/* at Fri Aug 01 10:56:14 2003
*/

```

```

/* Compiler settings for .\src\tpcc_com_ps.idl:
   oicf, w1, zp8, env=win32 (32b run)
   protocol : dce ,ms_ext, c_ext
   error checks: allocation ref bounds_check enum
stub_data
   VC __declspec() decoration level:
       __declspec(uuid()), __declspec(selectany),
       __declspec(novtable)
       DECLSPEC_UUID(), MIDL_INTERFACE()
*/
//@@MIDL_FILE_HEADING( )

#if !defined(_M_IA64) && !defined(_M_AMD64)
#define USE_STUBLESS_PROXY

/* verify that the <rpcproxy.h> version is high enough
to compile this file*/
#ifndef __REDQ_RPCPROXY_H_VERSION__
#define __REQUIRED_RPCPROXY_H_VERSION__ 440
#endif

#include "rpcproxy.h"
#ifdef __RPCPROXY_H_VERSION__
#error this stub requires an updated version of
<rpcproxy.h>
#endif // __RPCPROXY_H_VERSION__

#include "tpcc_com_ps.h"

#define TYPE_FORMAT_STRING_SIZE 1023
#define PROC_FORMAT_STRING_SIZE 193
#define TRANSMIT_AS_TABLE_SIZE 0
#define WIRE_MARSHAL_TABLE_SIZE 1

typedef struct _MIDL_TYPE_FORMAT_STRING
{
    short          Pad;
    unsigned char  Format[ TYPE_FORMAT_STRING_SIZE ];
} MIDL_TYPE_FORMAT_STRING;

typedef struct _MIDL_PROC_FORMAT_STRING
{
    short          Pad;
    unsigned char  Format[ PROC_FORMAT_STRING_SIZE ];
} MIDL_PROC_FORMAT_STRING;

static RPC_SYNTAX_IDENTIFIER _RpcTransferSyntax =
{{0x8a885d04,0x1ceb,0x11c9,{0x9f,0xe8,0x08,0x00,0x2b,0x
10,0x48,0x60}},{2,0}};

extern const MIDL_TYPE_FORMAT_STRING
__MIDL_TypeFormatString;
extern const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString;

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO ITPCC_ServerInfo;
extern const MIDL_STUBLESS_PROXY_INFO ITPCC_ProxyInfo;

extern const USER_MARSHAL_ROUTINE_QUADRUPLE
UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ];

#if !defined(__RPC_WIN32__)

```

```

#error Invalid build platform for this stub.
#endif

#if !(TARGET_IS_NT40_OR_LATER)
#error You need a windows NT 4.0 or later to run this
stub because it uses these features:
#error -oif or -oicf, [wire_marshal] or
[user_marshall] attribute.
#error However, your C/C++ compilation flags indicate
you intend to run this app on earlier systems.
#error This app will die there with the
RPC_X_WRONG_STUB_VERSION error.
#endif

static const MIDL_PROC_FORMAT_STRING
__MIDL_ProcFormatString =
{
    0,
    {
        /* Procedure NewOrder */

FC_AUTO_HANDLE */          0x33,          /*
Old Flags: object, Oi2 */
/* 2 */ NdrFcLong( 0x0 ), /* 0 */
/* 6 */ NdrFcShort( 0x3 ), /* 3 */
/* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset
= 28 */
/* 10 */ NdrFcShort( 0x0 ), /* 0 */
/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x7, /* Oi2 Flags: srv must
size, c1t must size, has return, */
/* 16 */ 0x3, /* 3
*/
/* Parameter txn_in */

/* 16 */ NdrFcShort( 0x8b ), /* Flags: must size,
must free, in, by val, */
/* 18 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset
= 4 */
/* 20 */ NdrFcShort( 0x3e2 ), /* Type
Offset=994 */
/* Parameter txn_out */

/* 22 */ NdrFcShort( 0x4113 ), /* Flags: must
size, must free, out, simple ref, srv alloc size=16 */
/* 24 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset
= 20 */
/* 26 */ NdrFcShort( 0x3f4 ), /* Type
Offset=1012 */
/* Return value */

/* 28 */ NdrFcShort( 0x70 ), /* Flags: out, return,
base type, */
/* 30 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset
= 24 */
/* 32 */ 0x8, /* FC_LONG */
/* 0
*/
/* Procedure Payment */

/* 34 */ 0x33, /* FC_AUTO_HANDLE */
/* 36 */ NdrFcLong( 0x0 ), /* 0 */

```

```

/* 40 */ NdrFcShort( 0x4 ), /* 4 */
/* 42 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset
= 28 */
/* 44 */ NdrFcShort( 0x0 ), /* 0 */
/* 46 */ NdrFcShort( 0x8 ), /* 8 */
/* 48 */ 0x7, /* Oi2 Flags: srv must
size, c1t must size, has return, */
/* 50 */ 0x3, /* 3
*/
/* Parameter txn_in */

/* 50 */ NdrFcShort( 0x8b ), /* Flags: must size,
must free, in, by val, */
/* 52 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset
= 4 */
/* 54 */ NdrFcShort( 0x3e2 ), /* Type
Offset=994 */
/* Parameter txn_out */

/* 56 */ NdrFcShort( 0x4113 ), /* Flags: must
size, must free, out, simple ref, srv alloc size=16 */
/* 58 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset
= 20 */
/* 60 */ NdrFcShort( 0x3f4 ), /* Type
Offset=1012 */
/* Return value */

/* 62 */ NdrFcShort( 0x70 ), /* Flags: out, return,
base type, */
/* 64 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset
= 24 */
/* 66 */ 0x8, /* FC_LONG */
/* 0
*/
/* Procedure Delivery */

/* 68 */ 0x33, /* FC_AUTO_HANDLE */
/* 70 */ 0x6c, /*
Old Flags: object, Oi2 */
/* 70 */ NdrFcLong( 0x0 ), /* 0 */
/* 74 */ NdrFcShort( 0x5 ), /* 5 */
/* 76 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset
= 28 */
/* 78 */ NdrFcShort( 0x0 ), /* 0 */
/* 80 */ NdrFcShort( 0x8 ), /* 8 */
/* 82 */ 0x7, /* Oi2 Flags: srv must
size, c1t must size, has return, */
/* 84 */ 0x3, /* 3
*/
/* Parameter txn_in */

/* 84 */ NdrFcShort( 0x8b ), /* Flags: must size,
must free, in, by val, */
/* 86 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset
= 4 */
/* 88 */ NdrFcShort( 0x3e2 ), /* Type
Offset=994 */
/* Parameter txn_out */

/* 90 */ NdrFcShort( 0x4113 ), /* Flags: must
size, must free, out, simple ref, srv alloc size=16 */
/* 92 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset
= 20 */
/* 94 */ NdrFcShort( 0x3f4 ), /* Type
Offset=1012 */

```

```

/* Return value */
/* 96 */ NdrFcShort( 0x70 ), /* Flags: out, return,
base type, */
/* 98 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset
= 24 */
/* 100 */ 0x8, /* FC_LONG */
/* 0 */
/* Procedure StockLevel */
/* 102 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /*
Old Flags: object, Oi2 */
/* 104 */ NdrFCLong( 0x0 ), /* 0 */
/* 108 */ NdrFcShort( 0x6 ), /* 6 */
/* 110 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset
= 28 */
/* 112 */ NdrFcShort( 0x0 ), /* 0 */
/* 114 */ NdrFcShort( 0x8 ), /* 8 */
/* 116 */ 0x7, /* Oi2 Flags: srv must
size, c1t must size, has return, */
/* 0x3, /* 3 */
/* Parameter txn_in */
/* 118 */ NdrFcShort( 0x8b ), /* Flags: must size,
must free, in, by val, */
/* 120 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset
= 4 */
/* 122 */ NdrFcShort( 0x3e2 ), /* Type
Offset=994 */
/* Parameter txn_out */
/* 124 */ NdrFcShort( 0x4113 ), /* Flags: must
size, must free, out, simple ref, srv alloc size=16 */
/* 126 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset
= 20 */
/* 128 */ NdrFcShort( 0x3f4 ), /* Type
Offset=1012 */
/* Return value */
/* 130 */ NdrFcShort( 0x70 ), /* Flags: out, return,
base type, */
/* 132 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset
= 24 */
/* 134 */ 0x8, /* FC_LONG */
/* 0 */
/* Procedure OrderStatus */
/* 136 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /*
Old Flags: object, Oi2 */
/* 138 */ NdrFCLong( 0x0 ), /* 0 */
/* 142 */ NdrFcShort( 0x7 ), /* 7 */
/* 144 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset
= 28 */
/* 146 */ NdrFcShort( 0x0 ), /* 0 */
/* 148 */ NdrFcShort( 0x8 ), /* 8 */
/* 150 */ 0x7, /* Oi2 Flags: srv must
size, c1t must size, has return, */
/* 0x3, /* 3 */
/* Parameter txn_in */

```

```

/* 152 */ NdrFcShort( 0x8b ), /* Flags: must size,
must free, in, by val, */
/* 154 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset
= 4 */
/* 156 */ NdrFcShort( 0x3e2 ), /* Type
Offset=994 */
/* Parameter txn_out */
/* 158 */ NdrFcShort( 0x4113 ), /* Flags: must
size, must free, out, simple ref, srv alloc size=16 */
/* 160 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset
= 20 */
/* 162 */ NdrFcShort( 0x3f4 ), /* Type
Offset=1012 */
/* Return value */
/* 164 */ NdrFcShort( 0x70 ), /* Flags: out, return,
base type, */
/* 166 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset
= 24 */
/* 168 */ 0x8, /* FC_LONG */
/* 0x0, /* 0 */
/* Procedure CallSetComplete */
/* 170 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /*
Old Flags: object, Oi2 */
/* 172 */ NdrFCLong( 0x0 ), /* 0 */
/* 176 */ NdrFcShort( 0x8 ), /* 8 */
/* 178 */ NdrFcShort( 0x8 ), /* x86 Stack size/offset
= 8 */
/* 180 */ NdrFcShort( 0x0 ), /* 0 */
/* 182 */ NdrFcShort( 0x8 ), /* 8 */
/* 184 */ 0x4, /* Oi2 Flags: has
return, */
/* 0x1, /* 1 */
/* Return value */
/* 186 */ NdrFcShort( 0x70 ), /* Flags: out, return,
base type, */
/* 188 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset
= 4 */
/* 190 */ 0x8, /* FC_LONG */
/* 0x0, /* 0 */
/* 0x0
};
static const MIDL_TYPE_FORMAT_STRING
__MIDL_TypeFormatString =
{
0,
{
NdrFcShort( 0x0 ), /* 0 */
/* 2 */
0x12, 0x0, /*
FC_UP */
/* 4 */ NdrFcShort( 0x3ca ), /* offset= 970
(974) */
/* 6 */
0x2b, /*
FC_NON_ENCAPSULATED_UNION */

```

```

FC_ULONG */
/* 8 */ 0x7, /* Corr desc: FC_USHORT
*/
0x0, /*
*/
/* 10 */ NdrFcShort( 0xffff ), /* -8 */
/* 12 */ NdrFcShort( 0x2 ), /* Offset= 2 (14) */
/* 14 */ NdrFcShort( 0x10 ), /* 16 */
/* 16 */ NdrFcShort( 0x2f ), /* 47 */
/* 18 */ NdrFCLong( 0x14 ), /* 20 */
/* 22 */ NdrFcShort( 0x800b ), /* simple arm
type: FC_HYPER */
/* 24 */ NdrFCLong( 0x3 ), /* 3 */
/* 28 */ NdrFcShort( 0x8008 ), /* simple arm
type: FC_LONG */
/* 30 */ NdrFCLong( 0x11 ), /* 17 */
/* 34 */ NdrFcShort( 0x8001 ), /* simple arm
type: FC_BYTE */
/* 36 */ NdrFCLong( 0x2 ), /* 2 */
/* 40 */ NdrFcShort( 0x8006 ), /* simple arm
type: FC_SHORT */
/* 42 */ NdrFCLong( 0x4 ), /* 4 */
/* 46 */ NdrFcShort( 0x800a ), /* simple arm
type: FC_FLOAT */
/* 48 */ NdrFCLong( 0x5 ), /* 5 */
/* 52 */ NdrFcShort( 0x800c ), /* simple arm
type: FC_DOUBLE */
/* 54 */ NdrFCLong( 0xb ), /* 11 */
/* 58 */ NdrFcShort( 0x8006 ), /* simple arm
type: FC_SHORT */
/* 60 */ NdrFCLong( 0xa ), /* 10 */
/* 64 */ NdrFcShort( 0x8008 ), /* simple arm
type: FC_LONG */
/* 66 */ NdrFCLong( 0x6 ), /* 6 */
/* 70 */ NdrFcShort( 0xe8 ), /* Offset= 232 (302) */
/* 72 */ NdrFCLong( 0x7 ), /* 7 */
/* 76 */ NdrFcShort( 0x800c ), /* simple arm
type: FC_DOUBLE */
/* 78 */ NdrFCLong( 0x8 ), /* 8 */
/* 82 */ NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 84 */ NdrFCLong( 0xd ), /* 13 */
/* 88 */ NdrFcShort( 0xf4 ), /* Offset= 244 (332) */
/* 90 */ NdrFCLong( 0x9 ), /* 9 */
/* 94 */ NdrFcShort( 0x100 ), /* Offset= 256
(350) */
/* 96 */ NdrFCLong( 0x2000 ), /* 8192 */
/* 100 */ NdrFcShort( 0x10c ), /* Offset= 268
(368) */
/* 102 */ NdrFCLong( 0x24 ), /* 36 */
/* 106 */ NdrFcShort( 0x31a ), /* Offset= 794
(900) */
/* 108 */ NdrFCLong( 0x4024 ), /* 16420 */
/* 112 */ NdrFcShort( 0x314 ), /* Offset= 788
(900) */
/* 114 */ NdrFCLong( 0x4011 ), /* 16401 */
/* 118 */ NdrFcShort( 0x312 ), /* Offset= 786
(904) */
/* 120 */ NdrFCLong( 0x4002 ), /* 16386 */
/* 124 */ NdrFcShort( 0x310 ), /* Offset= 784
(908) */
/* 126 */ NdrFCLong( 0x4003 ), /* 16387 */
/* 130 */ NdrFcShort( 0x30e ), /* Offset= 782
(912) */
/* 132 */ NdrFCLong( 0x4014 ), /* 16404 */
/* 136 */ NdrFcShort( 0x30c ), /* Offset= 780
(916) */
/* 138 */ NdrFCLong( 0x4004 ), /* 16388 */
/* 142 */ NdrFcShort( 0x30a ), /* Offset= 778
(920) */
/* 144 */ NdrFCLong( 0x4005 ), /* 16389 */

```

```

/* 148 */ NdrFcShort( 0x308 ), /* Offset= 776
(924) */
/* 150 */ NdrFcLong( 0x400b ), /* 16395 */
/* 154 */ NdrFcShort( 0x2f2 ), /* Offset= 754
(908) */
/* 156 */ NdrFcLong( 0x400a ), /* 16394 */
/* 160 */ NdrFcShort( 0x2f0 ), /* Offset= 752
(912) */
/* 162 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 166 */ NdrFcShort( 0x2fa ), /* Offset= 762
(928) */
/* 168 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 172 */ NdrFcShort( 0x2f0 ), /* Offset= 752
(924) */
/* 174 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 178 */ NdrFcShort( 0x2f2 ), /* Offset= 754
(932) */
/* 180 */ NdrFcLong( 0x400d ), /* 16397 */
/* 184 */ NdrFcShort( 0x2f0 ), /* Offset= 752
(936) */
/* 186 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 190 */ NdrFcShort( 0x2ee ), /* Offset= 750
(940) */
/* 192 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 196 */ NdrFcShort( 0x2ec ), /* Offset= 748
(944) */
/* 198 */ NdrFcLong( 0x400c ), /* 16396 */
/* 202 */ NdrFcShort( 0x2ea ), /* Offset= 746
(948) */
/* 204 */ NdrFcLong( 0x10 ), /* 16 */
/* 208 */ NdrFcShort( 0x8002 ), /* Simple arm
type: FC_CHAR */
/* 210 */ NdrFcLong( 0x12 ), /* 18 */
/* 214 */ NdrFcShort( 0x8006 ), /* Simple arm
type: FC_SHORT */
/* 216 */ NdrFcLong( 0x13 ), /* 19 */
/* 220 */ NdrFcShort( 0x8008 ), /* Simple arm
type: FC_LONG */
/* 222 */ NdrFcLong( 0x15 ), /* 21 */
/* 226 */ NdrFcShort( 0x800b ), /* Simple arm
type: FC_HYPER */
/* 228 */ NdrFcLong( 0x16 ), /* 22 */
/* 232 */ NdrFcShort( 0x8008 ), /* Simple arm
type: FC_LONG */
/* 234 */ NdrFcLong( 0x17 ), /* 23 */
/* 238 */ NdrFcShort( 0x8008 ), /* Simple arm
type: FC_LONG */
/* 240 */ NdrFcLong( 0xe ), /* 14 */
/* 244 */ NdrFcShort( 0x2c8 ), /* Offset= 712
(956) */
/* 246 */ NdrFcLong( 0x400e ), /* 16398 */
/* 250 */ NdrFcShort( 0x2cc ), /* Offset= 716
(966) */
/* 252 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 256 */ NdrFcShort( 0x2ca ), /* Offset= 714
(970) */
/* 258 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 262 */ NdrFcShort( 0x286 ), /* Offset= 646
(908) */
/* 264 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 268 */ NdrFcShort( 0x284 ), /* Offset= 644
(912) */
/* 270 */ NdrFcLong( 0x4015 ), /* 16405 */
/* 274 */ NdrFcShort( 0x282 ), /* Offset= 642
(916) */
/* 276 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 280 */ NdrFcShort( 0x278 ), /* Offset= 632
(912) */
/* 282 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 286 */ NdrFcShort( 0x272 ), /* Offset= 626
(912) */
/* 288 */ NdrFcLong( 0x0 ), /* 0 */

```

```

/* 292 */ NdrFcShort( 0x0 ), /* Offset= 0 (292) */
/* 294 */ NdrFcLong( 0x1 ), /* 1 */
/* 298 */ NdrFcShort( 0x0 ), /* Offset= 0 (298) */
/* 300 */ NdrFcShort( 0xffffffff ), /* Offset= -1
(299) */
/* 302 */
FC_STRUCT */
0x15, /*
0x7, /* 7
*/
/* 304 */ NdrFcShort( 0x8 ), /* 8 */
/* 306 */ 0xb, /* FC_HYPER */
0x5b, /*
FC_END */
/* 308 */
0x12, 0x0, /*
FC_UP */
/* 310 */ NdrFcShort( 0xc ), /* Offset= 12 (322) */
/* 312 */
0x1b, /*
FC_CARRAY */
0x1, /* 1
*/
/* 314 */ NdrFcShort( 0x2 ), /* 2 */
/* 316 */ 0x9, /* Corr desc: FC_ULONG */
0x0, /*
*/
/* 318 */ NdrFcShort( 0xfffc ), /* -4 */
/* 320 */ 0x6, /* FC_SHORT */
0x5b, /*
FC_END */
/* 322 */
0x17, /*
FC_CSTRUCT */
0x3, /* 3
*/
/* 324 */ NdrFcShort( 0x8 ), /* 8 */
/* 326 */ NdrFcShort( 0xffffffff ), /* Offset= -14
(312) */
/* 328 */ 0x8, /* FC_LONG */
0x8, /*
FC_LONG */
/* 330 */ 0x5c, /* FC_PAD */
0x5b, /*
FC_END */
/* 332 */
0x2f, /*
FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 334 */ NdrFcLong( 0x0 ), /* 0 */
/* 338 */ NdrFcShort( 0x0 ), /* 0 */
/* 340 */ NdrFcShort( 0x0 ), /* 0 */
/* 342 */ 0xc0, /* 192 */
0x0, /* 0
*/
/* 344 */ 0x0, /* 0 */
0x0, /* 0
*/
/* 346 */ 0x0, /* 0 */
0x0, /* 0
*/
/* 348 */ 0x0, /* 0 */
0x46, /* 70
*/
/* 350 */
0x2f, /*
FC_IP */
0x5a, /*
FC_CONSTANT_IID */
/* 352 */ NdrFcLong( 0x20400 ), /* 132096 */
/* 356 */ NdrFcShort( 0x0 ), /* 0 */

```

```

/* 358 */ NdrFcShort( 0x0 ), /* 0 */
/* 360 */ 0xc0, /* 192 */
0x0, /* 0
*/
/* 362 */ 0x0, /* 0 */
0x0, /* 0
*/
/* 364 */ 0x0, /* 0 */
0x0, /* 0
*/
/* 366 */ 0x0, /* 0 */
0x46, /* 70
*/
/* 368 */
0x12, 0x10, /*
FC_UP [pointer_deref] */
/* 370 */ NdrFcShort( 0x2 ), /* offset= 2 (372) */
/* 372 */
0x12, 0x0, /*
FC_UP */
/* 374 */ NdrFcShort( 0x1fc ), /* offset= 508
(882) */
/* 376 */
0x2a, /*
FC_ENCAPSULATED_UNION */
0x49, /* 73
*/
/* 378 */ NdrFcShort( 0x18 ), /* 24 */
/* 380 */ NdrFcShort( 0xa ), /* 10 */
/* 382 */ NdrFcLong( 0x8 ), /* 8 */
/* 386 */ NdrFcShort( 0x58 ), /* Offset= 88 (474) */
/* 388 */ NdrFcLong( 0xd ), /* 13 */
/* 392 */ NdrFcShort( 0x78 ), /* Offset= 120 (512) */
/* 394 */ NdrFcLong( 0x9 ), /* 9 */
/* 398 */ NdrFcShort( 0x94 ), /* Offset= 148 (546) */
/* 400 */ NdrFcLong( 0xc ), /* 12 */
/* 404 */ NdrFcShort( 0xbc ), /* Offset= 188 (592) */
/* 406 */ NdrFcLong( 0x24 ), /* 36 */
/* 410 */ NdrFcShort( 0x114 ), /* Offset= 276
(686) */
/* 412 */ NdrFcLong( 0x800d ), /* 32781 */
/* 416 */ NdrFcShort( 0x130 ), /* Offset= 304
(720) */
/* 418 */ NdrFcLong( 0x10 ), /* 16 */
/* 422 */ NdrFcShort( 0x148 ), /* offset= 328
(750) */
/* 424 */ NdrFcLong( 0x2 ), /* 2 */
/* 428 */ NdrFcShort( 0x160 ), /* offset= 352
(780) */
/* 430 */ NdrFcLong( 0x3 ), /* 3 */
/* 434 */ NdrFcShort( 0x178 ), /* offset= 376
(810) */
/* 436 */ NdrFcLong( 0x14 ), /* 20 */
/* 440 */ NdrFcShort( 0x190 ), /* offset= 400
(840) */
/* 442 */ NdrFcShort( 0xffffffff ), /* offset= -1
(441) */
/* 444 */
0x1b, /*
FC_CARRAY */
0x3, /* 3
*/
/* 446 */ NdrFcShort( 0x4 ), /* 4 */
/* 448 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0, /*
*/
/* 450 */ NdrFcShort( 0x0 ), /* 0 */
/* 452 */
0x4b, /*
FC_PP */

```



```

FC_PAD */
/* 454 */
FC_VARIABLE_REPEAT */
0x48,
0x49,
FC_FIXED_OFFSET */
/* 456 */ NdrFcShort( 0x4 ), /* 4 */
/* 458 */ NdrFcShort( 0x0 ), /* 0 */
/* 460 */ NdrFcShort( 0x1 ), /* 1 */
/* 462 */ NdrFcShort( 0x0 ), /* 0 */
/* 464 */ NdrFcShort( 0x0 ), /* 0 */
/* 466 */ 0x12, 0x0, /* FC_UP */
/* 468 */ NdrFcShort( 0xfffff6e ), /* offset= -146
(322) */
/* 470 */
FC_END */
0x5b,
FC_LONG */
/* 472 */ 0x5c, /* FC_PAD */
0x5b,
FC_END */
/* 474 */
FC_PSTRUCT */
0x16,
/* 476 */ NdrFcShort( 0x8 ), /* 8 */
/* 478 */
FC_PP */
0x4b,
FC_PAD */
/* 480 */
FC_NO_REPEAT */
0x46,
FC_PAD */
/* 482 */ NdrFcShort( 0x4 ), /* 4 */
/* 484 */ NdrFcShort( 0x4 ), /* 4 */
/* 486 */ 0x11, 0x0, /* FC_RP */
/* 488 */ NdrFcShort( 0xfffffd4 ), /* offset= -44
(444) */
/* 490 */
FC_END */
0x5b,
FC_LONG */
/* 492 */ 0x8, /* FC_LONG */
0x5b,
FC_END */
/* 494 */
FC_BOGUS_ARRAY */
0x21,
/* 496 */ NdrFcShort( 0x0 ), /* 0 */
/* 498 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0,
/* 500 */ NdrFcShort( 0x0 ), /* 0 */
/* 502 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 506 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0,
/* 508 */ NdrFcShort( 0xfffff50 ), /* offset= -176
(332) */
/* 510 */ 0x5c, /* FC_PAD */

```

```

FC_END */
/* 512 */
FC_BOGUS_STRUCT */
0x3,
/* 514 */ NdrFcShort( 0x8 ), /* 8 */
/* 516 */ NdrFcShort( 0x0 ), /* 0 */
/* 518 */ NdrFcShort( 0x6 ), /* offset= 6 (524) */
/* 520 */ 0x8, /* FC_LONG */
0x36,
FC_POINTER */
/* 522 */ 0x5c, /* FC_PAD */
0x5b,
FC_END */
/* 524 */
FC_RP */
/* 526 */ NdrFcShort( 0xfffffe0 ), /* offset= -32
(494) */
/* 528 */
FC_BOGUS_ARRAY */
0x3,
/* 530 */ NdrFcShort( 0x0 ), /* 0 */
/* 532 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0,
/* 534 */ NdrFcShort( 0x0 ), /* 0 */
/* 536 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 540 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0,
/* 542 */ NdrFcShort( 0xfffff40 ), /* offset= -192
(350) */
/* 544 */ 0x5c, /* FC_PAD */
0x5b,
FC_END */
/* 546 */
FC_BOGUS_STRUCT */
0x1a,
/* 548 */ NdrFcShort( 0x8 ), /* 8 */
/* 550 */ NdrFcShort( 0x0 ), /* 0 */
/* 552 */ NdrFcShort( 0x6 ), /* offset= 6 (558) */
/* 554 */ 0x8, /* FC_LONG */
0x36,
FC_POINTER */
/* 556 */ 0x5c, /* FC_PAD */
0x5b,
FC_END */
/* 558 */
FC_RP */
/* 560 */ NdrFcShort( 0xfffffe0 ), /* offset= -32
(528) */
/* 562 */
FC_CARRAY */
0x1b,
/* 564 */ NdrFcShort( 0x4 ), /* 4 */
/* 566 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0,
/* 568 */ NdrFcShort( 0x0 ), /* 0 */
/* 570 */

```

```

FC_PP */
0x4b,
FC_PAD */
/* 572 */
FC_VARIABLE_REPEAT */
0x48,
0x49,
FC_FIXED_OFFSET */
/* 574 */ NdrFcShort( 0x4 ), /* 4 */
/* 576 */ NdrFcShort( 0x0 ), /* 0 */
/* 578 */ NdrFcShort( 0x1 ), /* 1 */
/* 580 */ NdrFcShort( 0x0 ), /* 0 */
/* 582 */ NdrFcShort( 0x0 ), /* 0 */
/* 584 */ 0x12, 0x0, /* FC_UP */
/* 586 */ NdrFcShort( 0x184 ), /* offset= 388
(974) */
/* 588 */
FC_END */
0x5b,
FC_LONG */
/* 590 */ 0x5c, /* FC_PAD */
0x5b,
FC_END */
/* 592 */
FC_BOGUS_STRUCT */
0x3,
/* 594 */ NdrFcShort( 0x8 ), /* 8 */
/* 596 */ NdrFcShort( 0x0 ), /* 0 */
/* 598 */ NdrFcShort( 0x6 ), /* offset= 6 (604) */
/* 600 */ 0x8, /* FC_LONG */
0x36,
FC_POINTER */
/* 602 */ 0x5c, /* FC_PAD */
0x5b,
FC_END */
/* 604 */
FC_RP */
/* 606 */ NdrFcShort( 0xfffffd4 ), /* offset= -44
(562) */
/* 608 */
FC_IP */
0x5a,
FC_CONSTANT_IID */
/* 610 */ NdrFcLong( 0x2f ), /* 47 */
/* 614 */ NdrFcShort( 0x0 ), /* 0 */
/* 616 */ NdrFcShort( 0x0 ), /* 0 */
/* 618 */ 0xc0, /* 192 */
0x0,
/* 620 */ 0x0, /* 0 */
0x0,
/* 622 */ 0x0, /* 0 */
0x0,
/* 624 */ 0x0, /* 0 */
0x46, /* 70 */
/* 626 */
FC_CARRAY */
0x1b,
/* 628 */ NdrFcShort( 0x1 ), /* 1 */

```

```

/* 630 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0, /*
*/
/* 632 */ NdrFcShort( 0x4 ), /* 4 */
/* 634 */ 0x1, /* FC_BYTE */
0x5b, /*
FC_END */
/* 636 */
FC_BOGUS_STRUCT */
0x1a, /*
*/
0x3, /* 3
*/
/* 638 */ NdrFcShort( 0x10 ), /* 16 */
/* 640 */ NdrFcShort( 0x0 ), /* 0 */
/* 642 */ NdrFcShort( 0xa ), /* Offset= 10 (652) */
/* 644 */ 0x8, /* FC_LONG */
0x8, /*
FC_LONG */
/* 646 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0
*/
/* 648 */ NdrFcShort( 0xffffffff8 ), /* Offset= -40
(608) */
/* 650 */ 0x36, /* FC_POINTER */
0x5b, /*
FC_END */
/* 652 */
0x12, 0x0, /*
FC_UP */
/* 654 */ NdrFcShort( 0xffffffffe4 ), /* offset= -28
(626) */
/* 656 */
0x1b, /*
FC_CARRAY */
0x3, /* 3
*/
/* 658 */ NdrFcShort( 0x4 ), /* 4 */
/* 660 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0, /*
*/
/* 662 */ NdrFcShort( 0x0 ), /* 0 */
/* 664 */
0x4b, /*
FC_PP */
0x5c, /*
FC_PAD */
/* 666 */
0x48, /*
FC_VARIABLE_REPEAT */
0x49, /*
FC_FIXED_OFFSET */
/* 668 */ NdrFcShort( 0x4 ), /* 4 */
/* 670 */ NdrFcShort( 0x0 ), /* 0 */
/* 672 */ NdrFcShort( 0x1 ), /* 1 */
/* 674 */ NdrFcShort( 0x0 ), /* 0 */
/* 676 */ NdrFcShort( 0x0 ), /* 0 */
/* 678 */ 0x12, 0x0, /* FC_UP */
/* 680 */ NdrFcShort( 0xffffffffd4 ), /* Offset= -44
(636) */
/* 682 */
0x5b, /*
FC_END */
0x8, /*
FC_LONG */
/* 684 */ 0x5c, /* FC_PAD */
0x5b, /*
FC_END */
/* 686 */

```

```

FC_BOGUS_STRUCT */
0x1a, /*
*/
0x3, /* 3
*/
/* 688 */ NdrFcShort( 0x8 ), /* 8 */
/* 690 */ NdrFcShort( 0x0 ), /* 0 */
/* 692 */ NdrFcShort( 0x6 ), /* Offset= 6 (698) */
/* 694 */ 0x8, /* FC_LONG */
0x36, /*
FC_POINTER */
/* 696 */ 0x5c, /* FC_PAD */
0x5b, /*
FC_END */
/* 698 */
0x11, 0x0, /*
FC_RP */
/* 700 */ NdrFcShort( 0xffffffffd4 ), /* offset= -44
(656) */
/* 702 */
0x1d, /*
FC_SMFARRAY */
0x0, /* 0
*/
/* 704 */ NdrFcShort( 0x8 ), /* 8 */
/* 706 */ 0x1, /* FC_BYTE */
0x5b, /*
FC_END */
/* 708 */
0x15, /*
FC_STRUCT */
0x3, /* 3
*/
/* 710 */ NdrFcShort( 0x10 ), /* 16 */
/* 712 */ 0x8, /* FC_LONG */
0x6, /*
FC_SHORT */
/* 714 */ 0x6, /* FC_SHORT */
0x4c, /*
FC_EMBEDDED_COMPLEX */
/* 716 */ 0x0, /* 0 */
NdrFcShort( 0xfffffffff1 ),
/* offset= -15 (702) */
0x5b, /*
FC_END */
/* 720 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3
*/
/* 722 */ NdrFcShort( 0x18 ), /* 24 */
/* 724 */ NdrFcShort( 0x0 ), /* 0 */
/* 726 */ NdrFcShort( 0xa ), /* Offset= 10 (736) */
/* 728 */ 0x8, /* FC_LONG */
0x36, /*
FC_POINTER */
/* 730 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0
*/
/* 732 */ NdrFcShort( 0xffffffffe8 ), /* offset= -24
(708) */
/* 734 */ 0x5c, /* FC_PAD */
0x5b, /*
FC_END */
/* 736 */
0x11, 0x0, /*
FC_RP */
/* 738 */ NdrFcShort( 0xfffffffff0c ), /* offset= -244
(494) */
/* 740 */
0x1b, /*
FC_CARRAY */

```

```

0x0, /* 0
*/
/* 742 */ NdrFcShort( 0x1 ), /* 1 */
/* 744 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0, /*
*/
/* 746 */ NdrFcShort( 0x0 ), /* 0 */
/* 748 */ 0x1, /* FC_BYTE */
0x5b, /*
FC_END */
/* 750 */
0x16, /*
FC_PSTRUCT */
0x3, /* 3
*/
/* 752 */ NdrFcShort( 0x8 ), /* 8 */
/* 754 */
0x4b, /*
FC_PP */
0x5c, /*
FC_PAD */
/* 756 */
0x46, /*
FC_NO_REPEAT */
0x5c, /*
FC_PAD */
/* 758 */ NdrFcShort( 0x4 ), /* 4 */
/* 760 */ NdrFcShort( 0x4 ), /* 4 */
/* 762 */ 0x12, 0x0, /* FC_UP */
/* 764 */ NdrFcShort( 0xffffffffe8 ), /* offset= -24
(740) */
/* 766 */
0x5b, /*
FC_END */
0x8, /*
FC_LONG */
/* 768 */ 0x8, /* FC_LONG */
0x5b, /*
FC_END */
/* 770 */
0x1b, /*
FC_CARRAY */
0x1, /* 1
*/
/* 772 */ NdrFcShort( 0x2 ), /* 2 */
/* 774 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0, /*
*/
/* 776 */ NdrFcShort( 0x0 ), /* 0 */
/* 778 */ 0x6, /* FC_SHORT */
0x5b, /*
FC_END */
/* 780 */
0x16, /*
FC_PSTRUCT */
0x3, /* 3
*/
/* 782 */ NdrFcShort( 0x8 ), /* 8 */
/* 784 */
0x4b, /*
FC_PP */
0x5c, /*
FC_PAD */
/* 786 */
0x46, /*
FC_NO_REPEAT */
0x5c, /*
FC_PAD */
/* 788 */ NdrFcShort( 0x4 ), /* 4 */

```

```

/* 790 */ NdrFcShort( 0x4 ), /* 4 */
/* 792 */ 0x12, 0x0, /* FC_UP */
/* 794 */ NdrFcShort( 0xffffffff8 ), /* offset= -24
(770) */
/* 796 */
FC_END */
0x5b, /*
FC_LONG */
/* 798 */ 0x8, /* FC_LONG */
/* FC_UP */
/* 800 */
FC_END */
/* 800 */
FC_CARRAY */
0x1b, /*
0x3, /* 3
*/
/* 802 */ NdrFcShort( 0x4 ), /* 4 */
/* 804 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0, /*
*/
/* 806 */ NdrFcShort( 0x0 ), /* 0 */
/* 808 */ 0x8, /* FC_LONG */
0x5b, /*
FC_END */
/* 810 */
0x16, /*
FC_PSTRUCT */
0x3, /* 3
*/
/* 812 */ NdrFcShort( 0x8 ), /* 8 */
/* 814 */
0x4b, /*
FC_PP */
0x5c, /*
FC_PAD */
/* 816 */
0x46, /*
FC_NO_REPEAT */
0x5c, /*
FC_PAD */
/* 818 */ NdrFcShort( 0x4 ), /* 4 */
/* 820 */ NdrFcShort( 0x4 ), /* 4 */
/* 822 */ 0x12, 0x0, /* FC_UP */
/* 824 */ NdrFcShort( 0xffffffff8 ), /* offset= -24
(800) */
/* 826 */
0x5b, /*
FC_END */
0x8, /*
FC_LONG */
/* 828 */ 0x8, /* FC_LONG */
0x5b, /*
FC_END */
/* 830 */
0x1b, /*
FC_CARRAY */
0x7, /* 7
*/
/* 832 */ NdrFcShort( 0x8 ), /* 8 */
/* 834 */ 0x19, /* Corr desc: field
pointer, FC_ULONG */
0x0, /*
*/
/* 836 */ NdrFcShort( 0x0 ), /* 0 */
/* 838 */ 0xb, /* FC_HYPER */
0x5b, /*
FC_END */
/* 840 */

```

```

FC_PSTRUCT */
0x16, /*
0x3, /* 3
*/
/* 842 */ NdrFcShort( 0x8 ), /* 8 */
/* 844 */
0x4b, /*
FC_PP */
0x5c, /*
FC_PAD */
/* 846 */
0x46, /*
FC_NO_REPEAT */
0x5c, /*
FC_PAD */
/* 848 */ NdrFcShort( 0x4 ), /* 4 */
/* 850 */ NdrFcShort( 0x4 ), /* 4 */
/* 852 */ 0x12, 0x0, /* FC_UP */
/* 854 */ NdrFcShort( 0xffffffff8 ), /* offset= -24
(830) */
/* 856 */
0x5b, /*
FC_END */
0x8, /*
FC_LONG */
/* 858 */ 0x8, /* FC_LONG */
0x5b, /*
FC_END */
/* 860 */
0x15, /*
FC_STRUCT */
0x3, /* 3
*/
/* 862 */ NdrFcShort( 0x8 ), /* 8 */
/* 864 */ 0x8, /* FC_LONG */
0x8, /*
FC_LONG */
/* 866 */ 0x5c, /* FC_PAD */
0x5b, /*
FC_END */
/* 868 */
0x1b, /*
FC_CARRAY */
0x3, /* 3
*/
/* 870 */ NdrFcShort( 0x8 ), /* 8 */
/* 872 */ 0x7, /* Corr desc: FC_USHORT
*/
0x0, /*
*/
/* 874 */ NdrFcShort( 0xffd8 ), /* -40 */
/* 876 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0
*/
/* 878 */ NdrFcShort( 0xffffffffee ), /* offset= -18
(860) */
/* 880 */ 0x5c, /* FC_PAD */
0x5b, /*
FC_END */
/* 882 */
0x1a, /*
FC_BOGUS_STRUCT */
0x3, /* 3
*/
/* 884 */ NdrFcShort( 0x28 ), /* 40 */
/* 886 */ NdrFcShort( 0xffffffffee ), /* offset= -18
(868) */
/* 888 */ NdrFcShort( 0x0 ), /* offset= 0 (888) */
/* 890 */ 0x6, /* FC_SHORT */
0x6, /*
FC_SHORT */

```

```

/* 892 */ 0x8, /* FC_LONG */
0x8, /*
FC_LONG */
/* 894 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0
*/
/* 896 */ NdrFcShort( 0xfffffffffd8 ), /* offset= -520
(376) */
/* 898 */ 0x5c, /* FC_PAD */
0x5b, /*
FC_END */
/* 900 */
0x12, 0x0, /*
FC_UP */
/* 902 */ NdrFcShort( 0xfffffffffe6 ), /* offset= -266
(636) */
/* 904 */
FC_UP [simple_pointer] */
0x12, 0x8, /*
/* 906 */ 0x1, /* FC_BYTE */
0x5c, /*
FC_PAD */
/* 908 */
0x12, 0x8, /*
FC_UP [simple_pointer] */
/* 910 */ 0x6, /* FC_SHORT */
0x5c, /*
FC_PAD */
/* 912 */
FC_UP [simple_pointer] */
/* 914 */ 0x8, /* FC_LONG */
0x5c, /*
FC_PAD */
/* 916 */
0x12, 0x8, /*
FC_UP [simple_pointer] */
/* 918 */ 0xb, /* FC_HYPER */
0x5c, /*
FC_PAD */
/* 920 */
0x12, 0x8, /*
FC_UP [simple_pointer] */
/* 922 */ 0xa, /* FC_FLOAT */
0x5c, /*
FC_PAD */
/* 924 */
0x12, 0x8, /*
FC_UP [simple_pointer] */
/* 926 */ 0xc, /* FC_DOUBLE */
0x5c, /*
FC_PAD */
/* 928 */
0x12, 0x0, /*
FC_UP */
/* 930 */ NdrFcShort( 0xffffffffd8c ), /* offset= -628
(302) */
/* 932 */
0x12, 0x10, /*
FC_UP [pointer_deref] */
/* 934 */ NdrFcShort( 0xffffffffd8e ), /* offset= -626
(308) */
/* 936 */
0x12, 0x10, /*
FC_UP [pointer_deref] */
/* 938 */ NdrFcShort( 0xffffffffda2 ), /* offset= -606
(332) */
/* 940 */
0x12, 0x10, /*
FC_UP [pointer_deref] */
/* 942 */ NdrFcShort( 0xffffffffdb0 ), /* offset= -592
(350) */

```



```

const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[]
=
{
    ( CInterfaceStubVtbl * ) &ITPCStubVtbl,
    0
};

PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[]
=
{
    "ITPC",
    0
};

#define _tpcc_com_ps_CHECK_IID(n)
IID_GENERIC_CHECK_IID( _tpcc_com_ps, pIID, n)

int __stdcall _tpcc_com_ps_IID_Lookup( const IID *
pIID, int * pIndex )
{
    if(!_tpcc_com_ps_CHECK_IID(0))
    {
        *pIndex = 0;
        return 1;
    }
    return 0;
}

const ExtendedProxyFileInfo tpcc_com_ps_ProxyFileInfo =
{
    (PCInterfaceProxyVtblList *) &
    _tpcc_com_ps_ProxyVtblList,
    (PCInterfaceStubVtblList *) &
    _tpcc_com_ps_StubVtblList,
    (Const PCInterfaceName * ) &
    _tpcc_com_ps_InterfaceNamesList,
    0, // no delegation
    & _tpcc_com_ps_IID_Lookup,
    1,
    2,
    0, /* table of [async_uuid] interfaces */
    0, /* Filler1 */
    0, /* Filler2 */
    0, /* Filler3 */
};

#endif /* !defined(_M_IA64) && !defined(_M_AMD64)*/

```

common/txnlog/include/rtetime.h

```

/* FILE: rtetime.h : header file
* Copyright 1997 Microsoft Corp., All rights reserved.
* Source code licensed to Tandem Computers for
Internal
* use only. Redistribution of source or object files
or
* any derivative works is prohibited. By agreement,
this
* notice may not be removed.
* Authors: Charles Levine, Philip Durr

```

```

*
* Microsoft Corp.
*
*/
//FILE: RTETIME.H
#define MAX_JULIAN_TIME
0x7FFFFFFFFFFFFFFF
#define JULIAN_TIME __int64
#define TC_TIME DWORD
extern "C"
{
    BOOL InitJulianTime(LPSYSTEMTIME
lpInitTime);
    JULIAN_TIME GetJulianTime(void);
    DWORD MyTickCount(void);
    void GetJulianAndTC(JULIAN_TIME
*pJulian, DWORD *pTC);
    JULIAN_TIME ConvertTo64BitTime(int iYear, int
iMonth, int iday, int iHour, int iMinute, int iSecond);
    JULIAN_TIME Get64BitTime(LPSYSTEMTIME
lpInitTime);
    int JulianDay( int yr, int
mm, int dd );
    void JulianToTime(JULIAN_TIME julianTS,
int* yr, int* mm, int* dd, int *hh, int *mi, int *ss );
    void JulianToCalendar( int day, int* yr,
int* mm, int* dd );
}

```

common/txnlog/include/spinlock.h

```

/* FILE: SPINLOCK.H
* Copyright 1997 Microsoft Corp., All rights reserved.
* Source code licensed to Tandem Computers for
Internal
* use only. Redistribution of source or object files
or
* any derivative works is prohibited. By agreement,
this
* notice may not be removed.
* Authors: Mike Parkes, Charles Levine, Philip Durr
Microsoft Corp.
*/

#ifdef _INC_Spinlock
    const LONG LockClosed = 1;
    const LONG LockOpen = 0;
    /*****
    * Spinlock and Semaphore locking.
    * This class provides a very conservative
locking scheme.
    * The assumption behind the code is that
locks will be
    * held for a very short time. When a lock
is taken a memory
    * location is exchanged. All other
threads that want this
    *****/

```

```

* lock wait by spinning and sometimes
sleeping on a semaphore
* until it becomes free again. The only
other choice is not
* to wait at all and move on to do
something else. This
* module should normally be used in
conjunction with cache
* aligned memory in minimize cache line
misses.
*

```

```

*****
*****/

```

```

class Spinlock
{
    // Private data.
    HANDLE
Semaphore;
    volatile LONG
m_Spinlock;
    volatile LONG
waiting;

#ifdef _DEBUG
    // Counters for debugging
    volatile LONG
builds.
    volatile LONG
TotalLocks;
    volatile LONG
TotalSleeps;
    volatile LONG
TotalSpins;
    volatile LONG
TotalWaits;
#endif

```

```

public:
    // Public functions.
    Spinlock( void );
    inline BOOL claimLock(
void );
    inline void ReleaseLock(
~Spinlock( void );
// Disabled operations.
Spinlock( const Spinlock
& Copy );
    void operator=( const
Spinlock & Copy );

private:
    // Private functions.
    inline BOOL
ClaimSpinlock( volatile LONG *sl );
    void waitForLock( void );
    void wakeAllSleepers(
void );
};

```

```

/*****
*
* A guaranteed atomic exchange.
*
* An attempt is made to claim the
Spinlock. This action is
* guaranteed to be atomic.
*

```

```

*****
*****/
inline BOOL Spinlock::ClaimSpinlock( volatile
LONG *Spinlock )
{
    #ifdef _DEBUG
        InterlockedIncrement(
(LPLONG) & TotalLocks );
    #endif
    return ( ((*Spinlock) == LockOpen)
&& (InterlockedExchange( (LPLONG)Spinlock, LockClosed)
== LockOpen) );
}
/*****
*
* Claim the Spinlock.
* Claim the lock if available else wait or
exit.
*
*****/
inline BOOL Spinlock::ClaimLock( BOOL wait )
{
    if ( ! ClaimSpinlock( (volatile
LONG*) & m_Spinlock ) )
    {
        if ( wait )
            WaitForLock();
        return Wait;
    }
    return TRUE;
}
/*****
*
* Release the Spinlock.
* Release the lock and if needed wakeup
any sleepers.
*
*****/
inline void Spinlock::ReleaseLock( void )
{
    m_Spinlock = LockOpen;
    if ( waiting > 0 )
        wakeAllSleepers();
}
#define _INC_Spinlock
#endif
common/txnlog/include/txnlog.h
/* FILE: TXNLOG.H Microsoft TPC-C
Kit Ver. 4.10.000
* not yet audited
* PURPOSE: Header file for txn log class
* Copyright
Microsoft, 1999

```

```

* All Rights Reserved
*
*/
#include <stdio.h> //needed for FILE
#define DRIVER_NAME_LEN 32 //max length of the
driver engine name - must be the same as in engstut.h!
#define TXN_LOG_INCORRECTLY_SHUT_DOWN 100
//ctrl rec subtype generated by the txn log
when reading an abruptly shut down log
#pragma once
typedef struct _TXN_NEWORDER
{
    BYTE OL_Count; //range 0 to 31
    BYTE OL_Remote_Count; //range 0 to 31
    WORD C_id;
    int o_id;
} TXN_NEWORDER;
typedef struct _TXN_PAYMENT
{
    BYTE CustByName;
    BYTE ISRemote;
} TXN_PAYMENT;
typedef struct _TXN_ORDERSTATUS
{
    BYTE CustByName;
} TXN_ORDERSTATUS;
typedef union _TXN_DETAILS
{
    TXN_NEWORDER NewOrder;
    TXN_PAYMENT Payment;
    TXN_ORDERSTATUS OrderStatus;
} TXN_DETAILS;
// Common header for all records in txn log.
The TxnType field is
// a switch which identifies the particular
variant.
#define TXN_REC_TYPE_CONTROL 1
//
#define TXN_REC_TYPE_TPCC 2 // replaces TRANSACTION_TYPE_TPCC
#define TXN_REC_TYPE_TPCC_DELIV_DEF 3
#define TXN_REC_TYPE_TPCW 4 // replaces TRANSACTION_TYPE_TPCW
typedef struct _TXN_RECORD_HEADER
{
    JULIAN_TIME TxnStartT0;
    // start of txn
    BYTE TxnType;
    // one of TXN_REC_TYPE_*
    BYTE TxnSubType;
    // depends on TxnType
} TXN_RECORD_HEADER, *PTXN_RECORD_HEADER;
typedef struct _TXN_RECORD_CONTROL
{
    // common header; must exactly
match TXN_RECORD_HEADER
    JULIAN_TIME TxnStartT0;
    // start of txn

```

```

    BYTE TxnType;
    // = TXN_REC_TYPE_CONTROL
    BYTE TxnSubType;
    // depends on TxnType
    // end of common header
    DWORD Len;
    // number of bytes after this field
} TXN_RECORD_CONTROL, *PTXN_RECORD_CONTROL;
// TPC-C Txn Record Layout:
//
// 'TxnStartT0' is a Julian timestamp corresponding
to the moment the
// txn is sent to the SUT, i.e., beginning of
response time. Deltas
// are in milliseconds. Note that if RTDelay > 0,
then the txn was
// delayed by this amount. The delay occurs at the
beginning of the
// response time. So if RTDelay > 0, then the txn
was actually sent
// at TxnStartT0 + RTDelay.
// Graphically:
//
// time -->
//
// |--- Menu ---|--- Keying ---|--- Response ---|
-- Think --|
// <- DeltaT1 -> <- DeltaT2 -> <- DeltaT4 ->
<- DeltaT3 ->
//
// ^
// ^ TxnStartT0
// RTDelay is the amount of response time delay
included in DeltaT4.
// RTDelay is recorded per txn because this value
can be changed on
// the fly, and so may vary from txn to txn.
// TxnStatus is the txn completion code. It is used
to indicate errors.
// For example, in the New Order txn, 1% of txns
abort. TxnStatus will
// reflect this.
typedef struct _TXN_RECORD_TPCC
{
    // common header; must exactly
match TXN_RECORD_HEADER
    JULIAN_TIME TxnStartT0;
    // start of txn
    BYTE TxnType;
    // = TXN_REC_TYPE_TPCC
    BYTE TxnSubType;
    // depends on TxnType
    // end of common header
    int DeltaT1; //
    int DeltaT2; //
    int DeltaT3; //
    int DeltaT4; //
    int RTDelay; //
    int response time delay (ms) //
}

```

```

    int TxnError;
    // error code providing more detail for
TxnStatus
    int w_id;
    // warehouse ID
    BYTE d_id;
    // assigned district ID for this thread
    BYTE d_id_ThisTxn; //
district ID chosen for this particular
    BYTE TxnStatus;
errors // completion status for txn to indicate
    BYTE reserved; //
for word alignment
    TXN_DETAILS TxnDetails;
//
bool IsSuccessRecord() { return
(TxnStatus == ERR_SUCCESS || TxnStatus ==
ERR_BAD_ITEM_ID || TxnStatus ==
ERR_TYPE_DELIVERY_POST); }
} TXN_RECORD_TPCC, *PTXN_RECORD_TPCC;

// TPC-C Deferred Delivery Txn Record Layout:
//
//Incorporating delivery transaction information
into the above
//structure would increase the size of
TXN_DETAILS from 8 to 42 bytes.
//Hence, we store delivery transaction details in a
separate structure.
//
typedef struct _TXN_RECORD_TPCC_DELIV_DEF
{
// common header; must exactly
match TXN_RECORD_HEADER
    JULIAN_TIME TxnStartT0;
    // start of txn
    BYTE TxnType;
    // = TXN_REC_TYPE_TPCC_DELIV_DEF
    BYTE TxnSubType;
    // = 0
    // end of common header
    int DeltaT4; //
response time (ms)
    int DeltaTxnExec; //
// execution time (ms)
    int w_id;
    // warehouse ID
    BYTE TxnStatus;
errors // completion status for txn to indicate
    BYTE reserved; //
for word alignment
    short o_carrier_id; //
carrier id
    long o_id[10]; //
returned delivery transaction ids
    bool IsSuccessRecord() { return
(TxnStatus == ERR_SUCCESS || TxnStatus ==
ERR_BAD_ITEM_ID || TxnStatus ==
ERR_TYPE_DELIVERY_POST); }
} TXN_RECORD_TPCC_DELIV_DEF,
*PTXN_RECORD_TPCC_DELIV_DEF;

//TPC-W records.
//
typedef struct _TXN_RECORD_TPCW
{

```

```

// common header; must exactly
match TXN_RECORD_HEADER
    JULIAN_TIME TxnStartT0;
    // start of txn
    BYTE TxnType;
    // = TXN_REC_TYPE_TPCW
    BYTE TxnSubType;
    // depends on TxnType
    // end of common header
    int ThinkTime; //
think time (ms)
    int WIRT;
    // response time (ms)
    int TxnError;
TxnStatus // error code providing more detail for
    BYTE TxnStatus;
errors // completion status for txn to indicate
    //this field below depends on the
txn sub type:
    // - for Home interaction: it
indicates whether the user was a new customer (or
returning)
    // - for Buy Confirm:
it indicates whether the shipping address was
updated
    // - for Search Request:
it indicates the search type (Author, Title,
or Subject)
    //this statistics needs to be
reported according to 5.5.5.1 clause in the specs.
    //because this field occupies 1
byte, the record structure is already aligned on word
boundary.
    union {
        BYTE newCustomer;
        BYTE addrUpdated;
        BYTE searchType;
    } intrDetails;
//This field is mostly for
informational/debugging purposes.
//It indicates what user performed
this web interaction and what instance (session) of
that use it was.
    //The first 22 bits indicate the
user #, and the top 10 bits indicate instance (session)
#.
    unsigned __int32 uiUser;
    bool IsSuccessRecord() { return
(TxnStatus == ERR_SUCCESS); }
} TXN_RECORD_TPCW, *PTXN_RECORD_TPCW;

//
// Data part of a control record
written when a user is created (or it's new session) -
to record
USMD
typedef struct _TXN_RECORD_TPCW_USER_DATA
{
    unsigned __int32 uiUser;
    JULIAN_TIME //
    USMD;
USMD for this user
    BYTE
    bRetCust; // returning
customer?
} TXN_RECORD_TPCW_USER_DATA,
*PTXN_RECORD_TPCW_USER_DATA;

```

```

//The entire TPCW user control record
structure
typedef struct _TXN_RECORD_TPCW_USER
{
// common header; must exactly
match TXN_RECORD_HEADER
    JULIAN_TIME TxnStartT0;
    // start of txn
    BYTE TxnType;
    // = TXN_REC_TYPE_CONTROL
    BYTE TxnSubType;
    // depends on TxnType
    // end of common header
    DWORD Len;
    // number of bytes after this field
    //The fields above must exactly
match TXN_RECORD_CONTROL
    //The fields below must exactly
match TXN_RECORD_TPCW_USER_DATA
    unsigned __int32 uiUser;
    // user number
    JULIAN_TIME
    USMD;
USMD for this user
    BYTE
    bRetCust; // returning
customer?
} TXN_RECORD_TPCW_USER,
*PTXN_RECORD_TPCW_USER;

#define USER_INDEX_NBITS 22
#define USER_INDEX_MASK //lower 22 bits
0x003fffff
mask for user field in TPCW record
#define USER_SESSION_MASK 0xffc00000
//upper 10 bits mask for user field
in TPCW record
#define USER_CREATE_REC 254
//subtype for the control record
written when a user is created
#define TXN_LOG_VERSION 2
#define TXN_DATA_START 4096
// offset
in log file where log records start
#define TXN_LOG_EYE_CATCHER "bc" //
signature bytes at the start of log file

////////////////////////////////////
// The transaction log has a header as the
first 4k block.
//
typedef struct _TXN_LOG_HEADER
{
    char
    EyeCatcher[2]; // signature bytes;
should always be "bc"
    int
    LogVersion; // set to
TXN_LOG_VERSION
    JULIAN_TIME
    BeginTxnTs; // timestamp of
first (lowest) txn start
    JULIAN_TIME
    EndTxnTs; // timestamp of last
(highest) txn completion time

```



```

        BYTE
**recPointers; //record pointer buffers
for each sorted block
    PTXN_RECORD_HEADER *recBuffers;
//record buffers for each sorted block
    int *PointersRead;
//# of pointers
processed in each block
    BOOL *BlockAvailable;
//whether to check a particular
block for jmin
    int nBlocks;
    int jmin;
record //index (block-wise) of the lowest timestamp
    int iAvgRecordLen;
//average
record length
    int iSortedReturnedCount;
//keeps track of the # of sorted records
returned through GetSortedRecord()
    BOOL bIncorrectShutDown;
// indicates whether the log opened
for read was not correctly shut down
    int write(BYTE *ptr, DWORD Size);
static void LogFileIO(CTxnLog *);
void LoadBuffers(int j);
//used in sort/merge to load record
buffers
static void
CheckpointThread(CTxnLog *); // checkpointing thread
public:
    CTxnLog(LPCTSTR szFileName, DWORD
dwOpts, char *szDriver = NULL);
~CTxnLog(void);
    int writeToLog(PTXN_RECORD_TPCC
pTxnRcrd);
    int writeToLog(PTXN_RECORD_TPCC_DELIV_DEF pTxnRcrd);
    int writeToLog(PTXN_RECORD_CONTROL
pCtrlRec);
    int writeToLog(PTXN_RECORD_HEADER
pCtrlRec);
    int writeToLog(PTXN_RECORD_TPCW
pTxnRcrd);
//support for TPC-W
    int writeCtrlRecToLog(BYTE SubType,
LPTSTR lpStr, DWORD
dwLen);
void CloseTransactionLogFile(void);
    PTXN_RECORD_HEADER
getNextRecord(BOOL bSkipCtrlRecs = FALSE);
    PTXN_RECORD_HEADER
getNextRecord(JULIAN_TIME SeekTimeT0, BOOL
bSkipCtrlRecs = FALSE);
    int Sort(void);
    PTXN_RECORD_HEADER
GetSortedRecord();

```

```

blogSorted; }; inline BOOL IsSorted(void) { return
return BeginTxnTS; }; inline JULIAN_TIME BeginTS(void) {
return EndTxnTS; }; inline JULIAN_TIME EndTS(void) {
return iRecCount; }; inline int RecordCount(void) {
};
class CTXNLOG_ERR : public CBaseErr
{
public:
    enum CTXNLOG_ERRS
    {
        ERR_BAD_FILE_FORMAT,
// "File format is invalid."
        ERR_UNKNOWN_LOG_VERSION,
// "Log file version is unknown."
        ERR_BROKEN_LOG_FILE,
// "Log file is broken."
        ERR_LOG_NOT_SORTED,
// "Log file is not sorted"
        ERR_INVALID_TIME_SEQ,
// "Internal Error: Record Time
Sequence invalid."
    };
    CTXNLOG_ERR(int iErr) :
CBaseErr(iErr) {};
    int ErrorType() {return
ERR_TYPE_TXNLOG;};
    char *ErrorTypeStr() { return "TXN
LOG"; }
    char *ErrorText()
    {
        static char *szMsgs[] = {
            "File format is
invalid.",
            "Log file
version is unknown.",
            "Log file is
broken.",
            "Log file is
not sorted",
            "Internal
Error: Record Time Sequence invalid.",
            ""
        };
        for(int i = 0;
szMsgs[i][0]; i++)
        {
            if ( m_idMsg ==
i )
                break;
        }
        return(szMsgs[i][0] ?
szMsgs[i] : ERR_UNKNOWN);
    };
};

```

Appendix B: Database Design

Build Scripts

setup.cmd

```
-----
:-- FILE:      RUNSQLCFG.CMD
:--           Microsoft TPC-C Kit Ver. 4.41
:--           Copyright Microsoft, 2001
:--           All Rights Reserved
:--
:-- PURPOSE:   Calls RunSQLCfg.sql to configure SQL
Server
:--
:-- ARGUMENTS: Optionally, the user can pass the
following positional arguments:
:--           Server Name
:--           sa SQL Server account password
:--           Number of Warehouses
:--           Build Option
:--
{full,builddb,objects,objectsfull,bulkload,bulkloadfull
,backup}
:--           Database Type
:--           {normal or scale_down}
:--
:--           If they are not passed, then the
user will be prompted by the VBS file.
:--
-----
@cscript SetupScripts\setup.vbs //H:CScript //I %1 %2
%3 %4 %5
```

createdb.sql

```
-----
--
-- File:      CREATEDB.SQL
--
--           Microsoft TPC-C Benchmark Kit Ver.
4.50
--           Copyright Microsoft, 2003
--
--
--           Creates 3520 warehouse database
--
```

```
--
-----
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(), 21))
go
-----
-- create main database files
CREATE DATABASE tpcc
ON PRIMARY
(
NAME = MSSQL_tpcc_root,
FILENAME = 'C:\MSSQL_tpcc_root.mdf',
SIZE = 8MB,
FILEGROWTH = 0),
FILEGROUP MSSQL_misc_fg
(
NAME = MSSQL_misc1,
FILENAME = 'E:',
SIZE = 52600MB,
FILEGROWTH = 0),
(
NAME = MSSQL_misc2,
FILENAME = 'F:',
SIZE = 52600MB,
FILEGROWTH = 0),
FILEGROUP MSSQL_cs_fg
(
NAME = MSSQL_cs1,
FILENAME = 'G:',
SIZE = 100000MB,
FILEGROWTH = 0),
(
NAME = MSSQL_cs2,
FILENAME = 'H:',
SIZE = 100000MB,
FILEGROWTH = 0)
LOG ON
(
NAME = MSSQL_tpcc_log,
FILENAME = 'L:',
SIZE = 135000MB,
FILEGROWTH = 0)
COLLATE Latin1_General_BIN
go
-----
-- Store ending time
update tpcc_timer
set end_date = (select convert(char(30),
getdate(), 21))
```

```
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
```

tables.sql

```
-- File:      TABLES.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.22
--           Copyright Microsoft, 2001
-- 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 MSSQL_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 MSSQL_misc_fg
go

create table customer
(
    c_id          int,
    c_d_id       tinyint,
    c_w_id       smallint,
    c_first      char(16),
    c_middle     char(16),
    c_last       char(16),
    c_street_1   char(20),
    c_street_2   char(20),
    c_city       char(20),

```

```

    c_state      char(2),
    c_zip        char(9),
    c_phone     char(16),
    c_since     datetime,
    c_credit     char(2),
    c_credit_lim numeric(12,2),
    c_discount   numeric(4,4),
    c_balance    numeric(12,2),
    c_ytd_payment numeric(12,2),
    c_payment_cnt smallint,
    c_delivery_cnt smallint,
    c_data      char(500)
) on MSSQL_cs_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 MSSQL_misc_fg
go

create table new_order
(
    no_o_id      int,
    no_d_id      tinyint,
    no_w_id      smallint,
    no_data     char(50)
) on MSSQL_misc_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 MSSQL_misc_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 MSSQL_misc_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 MSSQL_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 MSSQL_cs_fg
go

idxcuscl.sql

-- File:      IDXCUSCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.41
--           Copyright Microsoft, 2001
-- 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 MSSQL_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. 4.41
--           Copyright Microsoft, 2001
-- 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 MSSQL_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. 4.41
--           Copyright Microsoft, 2001
-- 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 MSSQL_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. 4.41
--           Copyright Microsoft, 2001
-- 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 MSSQL_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. 4.41
--           Copyright Microsoft, 2001
-- 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 MSSQL_misc_fg

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

```

idxodlcl.sql

```

-- File:      IDXODLCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.41
--           Copyright Microsoft, 2001
-- 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 MSSQL_misc_fg

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

```

idxordcl.sql

```

-- File:      IDXORDCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.41
--           Copyright Microsoft, 2001
-- 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 MSSQL_misc_fg

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

go

```

idxstkcl.sql

```

-- File:      IDXSTKCL.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- 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 MSSQL_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. 4.41
-- Copyright Microsoft, 2001
-- 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 MSSQL_misc_fg

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

go

```

idxordnc.sql

```

-- File:      IDXORDNC.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- Purpose:   Creates non-clustered index on orders
table

```

```

use tpcc
go

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

if exists ( select name from sysindexes where name =
'orders_nc1' )
drop index orders.orders_nc1

create index orders_nc1 on orders(o_w_id, o_d_id,
o_c_id, o_id)
on MSSQL_misc_fg

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

go

```

dbopt1.sql

```

-- File:      DBOPT1.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- 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
exec sp_dboption tpcc,'torn page detection',false
go

use tpcc
go

checkpoint
go

```

dbopt2.sql

```

-- File:      DBOPT2.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- Purpose:   Resets database options after data load

```

```

exec sp_dboption tpcc,'select into/bulkcopy',false
exec sp_dboption tpcc,'trunc. log on chkpt.',false
exec sp_dboption tpcc,'torn page detection',false
go

```

```

USE tpcc
GO

```

```

CHECKPOINT
GO

```

```

sp_configure 'allow updates',1
GO

```

```

RECONFIGURE WITH OVERRIDE
GO

```

```

DECLARE @msg varchar(50)

```

```

--
-- OPTIONS FOR SQL SERVER 2000
-- Set option values for user-defined indexes
--

```

```

SET @msg = ' '
PRINT @msg
SET @msg = 'Setting SQL Server 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
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

```

backup.sql

```

-- File:      BACKUP.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- Purpose:   Creates backup of tpcc database

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

dump database tpcc to tpccback1, tpccback2 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

```

restore.sql

```

-- File:      RESTORE.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- Purpose:   Loads database backup from backup files

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

load database tpcc from tpccback1, tpccback2 with stats
= 1

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

```

Stored Procedures

neword.sql

```

-- File:      NEWORD.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- Purpose:   Creates new order transaction stored
procedure
--
--      Interface Level: 4.10.000

use tpcc
go

if exists ( select name from sysobjects where name =
'neword' )
drop procedure neword

create proc neword
@w_id          int,
@o1_qty       tinyint,
@d_id         tinyint,
@c_id         int,
@o1_cnt       tinyint,

```

```

@o_all_local  tinyint,
@o1_qty1     @i_id1 int = 0, @s_w_id1 smallint = 0,
             smallint = 0,
@o1_qty2     @i_id2 int = 0, @s_w_id2 smallint = 0,
             smallint = 0,
@o1_qty3     @i_id3 int = 0, @s_w_id3 smallint = 0,
             smallint = 0,
@o1_qty4     @i_id4 int = 0, @s_w_id4 smallint = 0,
             smallint = 0,
@o1_qty5     @i_id5 int = 0, @s_w_id5 smallint = 0,
             smallint = 0,
@o1_qty6     @i_id6 int = 0, @s_w_id6 smallint = 0,
             smallint = 0,
@o1_qty7     @i_id7 int = 0, @s_w_id7 smallint = 0,
             smallint = 0,
@o1_qty8     @i_id8 int = 0, @s_w_id8 smallint = 0,
             smallint = 0,
@o1_qty9     @i_id9 int = 0, @s_w_id9 smallint = 0,
             smallint = 0,
@o1_qty10    @i_id10 int = 0, @s_w_id10 smallint = 0,
             smallint = 0,
@o1_qty11    @i_id11 int = 0, @s_w_id11 smallint = 0,
             smallint = 0,
@o1_qty12    @i_id12 int = 0, @s_w_id12 smallint = 0,
             smallint = 0,
@o1_qty13    @i_id13 int = 0, @s_w_id13 smallint = 0,
             smallint = 0,
@o1_qty14    @i_id14 int = 0, @s_w_id14 smallint = 0,
             smallint = 0,
@o1_qty15    @i_id15 int = 0, @s_w_id15 smallint = 0,
             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,
@li_id           int,
@commit_flag    tinyint,
@li_id          int,
@li_s_w_id      smallint,
@li_qty         smallint,
@ol_number      int,
@c_id_local     int

```

```

begin
begin transaction n
-- get district tax and next available order id and
update
-- plus initialize local variables
      update  district
      set     @d_tax      = d_tax,
             @o_id      = d_next_o_id,
             d_next_o_id = d_next_o_id + 1,
             @o_entry_d = getdate(),
             @li_no     = 0,
             @commit_flag = 1,
             d_w_id     = @w_id and
             d_id       = @d_id

-- process orderlines
      while (@li_no < @o_o1_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

                @li_qty = case @li_no
                when 1 then
@o1_qty1
                when 2 then
@o1_qty2
                when 3 then
@o1_qty3
                when 4 then
@o1_qty4
                when 5 then
@o1_qty5
                when 6 then
@o1_qty6
                when 7 then
@o1_qty7
                when 8 then
@o1_qty8
                when 9 then
@o1_qty9
                when 10 then
@o1_qty10
                when 11 then
@o1_qty11
                when 12 then
@o1_qty12
                when 13 then
@o1_qty13
                when 14 then
@o1_qty14
                when 15 then
@o1_qty15

-- get item data (no one updates item)
                select  @i_price = i_price,
                       @i_name  = i_name,
                       @i_data  = i_data
                from     item (tablock)
                where    i_id = @li_id

-- update stock values
                update  stock
                set     s_ytd =
s_ytd + @li_qty,

```

```

                @s_quantity =
s_quantity - @li_qty +
                case when (s_quantity - @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

```

payment.sql

```

-- File:      PAYMENT.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- 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),
        @data      char(500),
        @c_data    char(500),
        @datetime  datetime,
        @w_ytd    numeric(12,2),
        @d_ytd    numeric(12,2),
        @cnt      smallint,
        @val      smallint,
        @screen_data char(200),
        @d_id_local tinyint,
        @w_id_local smallint,
        @c_id_local int
select @screen_data = ''
begin tran p
-- get payment date
    select    @datetime = getdate()
    if (@c_id = 0)
        begin
-- get customer id and info using last name
        select    @cnt    = count(*)
        from      customer (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,
        @data      = c_data,

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

-- if customer has bad credit get some more info
    if (@c_credit = 'BC')
    begin

--      compute new info
    select @c_data =
    convert(char(5),@c_id) +
    convert(char(4),@c_d_id) +
    convert(char(5),@c_w_id) +
    convert(char(4),@d_id) +
    convert(char(5),@w_id) +
    convert(char(19),@h_amount) +
    substring(@data, 1, 458)

-- update customer info
        update customer
        set c_data = @c_data
    where
        c_id = @c_id and
        c_w_id = @c_w_id and
        c_d_id = @c_d_id

    select @screen_data = substring
    (@c_data,1,200)
    end

-- get district data and update year-to-date
        update district
        set d_ytd = d_ytd +
    @h_amount,
        @d_street_1 = d_street_1,
        @d_street_2 = d_street_2,
        @d_city     = d_city,
        @d_state   = d_state,
        @d_zip     = d_zip,
        @d_name    = d_name,
        @d_id_local = d_id
    where
        d_w_id = @w_id and
        d_id   = @d_id

-- get warehouse data and update year-to-date

```

```

        update warehouse
        set w_ytd = w_ytd +
    @h_amount,
        @w_street_1 = w_street_1,
        @w_street_2 = w_street_2,
        @w_city     = w_city,
        @w_state   = w_state,
        @w_zip     = w_zip,
        @w_name    = w_name,
        @w_id_local = w_id
    where
        w_id = @w_id

-- create history record
    insert into history values ( @c_id_local,
    @c_d_id,
    @c_w_id,
    @d_id_local,
    @w_id_local,
    @datetime,
    @h_amount,
    @w_name + ' ' + @d_name)
    commit tran p

-- return data to client
    select @c_id,
        @c_last,
        @datetime,
        @w_street_1,
        @w_street_2,
        @w_city,
        @w_state,
        @w_zip,
        @d_street_1,
        @d_street_2,
        @d_city,
        @d_state,
        @d_zip,
        @c_first,
        @c_middle,
        @c_street_1,
        @c_street_2,
        @c_city,
        @c_state,
        @c_zip,
        @c_phone,
        @c_since,
        @c_credit,
        @c_credit_lim,
        @c_discount,
        @c_balance,
        @screen_data

    go

ordstat.sql

-- File:   ORDSTAT.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41

```

```

-- Copyright Microsoft, 2001
-- 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 int,
@cnt int,
@o_id int,
@o_entry_d datetime,
@o_carrier_id smallint,
@cnt smallint
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
        (count(*)+1)/2
    select @cnt =
    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_balance =
    @c_first =
    @c_last =
    @c_middle =
    from customer (repeatableread)
    where c_last =
    @c_w_id =
    @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
c_balance,
c_last
@c_id and
@d_id and
@w_id
select
@c_balance =
@c_first = c_first,
@c_middle = c_middle,
@c_last =
from customer (repeatable read)
where c_id =
c_d_id =
c_w_id =
select @cnt = @@rowcount
end
-- if no such customer
if (@cnt = 0)
begin
found',18,1)
raiserror('Customer not
found',18,1)
goto custnotfound
end
-- get order info
select
@o_id = o_id,
@o_entry_d = o_entry_d,
@o_carrier_id = o_carrier_id
from orders (serializable)
where
o_c_id = @c_id and
o_d_id = @d_id and
o_w_id = @w_id
order
by o_id asc
-- select order lines for the current order
select
ol_supply_w_id,
ol_i_id,
ol_quantity,
ol_amount,
ol_delivery_d
from order_line (repeatable read)
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
delivery.sql
-- File: DELIVERY.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.41
-- Copyright Microsoft, 2001
-- 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
new_order (serializable)
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 =
no_d_id =
no_o_id =
@o_id
-- set carrier_id on this order (and get customer id)
update orders
set o_carrier_id
= @o_carrier_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
where
ol_w_id
= @w_id and
ol_d_id
= @d_id and
ol_o_id
= @o_id
-- accumulate lineitem amounts for this order into
customer
update customer
set c_balance =
c_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

```

stocklev.sql

```

-- File:      STOCKLEV.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.41
--           Copyright Microsoft, 2001
-- Purpose:   Creates stock level transaction stored
--           procedure
--           Interface Level: 4.10.000
use tpcc
go
if exists (select name from sysobjects where name =
'tpcc_stocklevel' )
drop procedure tpcc_stocklevel
go
create proc tpcc_stocklevel @w_id
smallint,
                        @d_id
tinyint,
                        @threshold
smallint
as
declare @o_id_low int,
        @o_id_high int
select @o_id_low = (d_next_o_id - 20),
       @o_id_high = (d_next_o_id - 1)
from district
where d_w_id = @w_id and
      d_id = @d_id
select count(distinct(s_i_id))
from stock, order_line
where o1_w_id = @w_id and
      o1_d_id = @d_id and
      o1_o_id between @o_id_low and
                @o_id_high and
      s_w_id = o1_w_id and
      s_i_id = o1_i_id and

```

```

s_quantity < @threshold
go

```

Loader Source Code

tpcc.h

```

// File:      TPCC.H      Microsoft TPC-C
// Kit Ver. 4.41
//           Copyright
//           Microsoft, 1996, 1997, 1998, 1999, 2000, 2001
// Purpose:   Header file for TPC-C database
loader

// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.41"

// 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 ""
#define PASSWORD "sa"

// Default loader arguments
#define BATCH 10000
#define DEF_LD_PACKSIZE 32768
#define LOADER_RES_FILE "C:\\MSTPCC.440\\SETUP\\logs\\load.out"
#define LOG_PATH "C:\\MSTPCC.440\\SETUP\\LOGS\\";
#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 *log_path;
    char *synch_servername;
    long case_sensitivity;
    long starting_warehouse;
    long build_index;
    long index_order;
    long scale_down;
    char *index_script_path;
} TPCC_LD_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: TPCCCLR.C Microsoft TPC-C
// Kit Ver. 4.41
// Copyright
// Microsoft, 1996, 1997, 1998, 1999, 2000, 2001
// 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 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 LoadOrderTable();
void LoadNewOrderTable();
void LoadOrderLineTable();
void GetPermutation();
void CheckForCommit();
void OpenConnections();
void BuildIndex();
void FormatDate ();

// Shared memory structures
typedef struct
{
    long ol;
    long ol_i_id;
    short ol_supply_w_id;
    short ol_quantity;
    double ol_amount;
    char ol_dist_info[DIST_INFO_LEN+1];
    char ol_delivery_d[OL_DELIVERY_D_LEN+1];
} ORDER_LINE_STRUCT;

typedef struct
{
    long o_id;
    short o_d_id;
    short o_w_id;
    long o_c_id;
    short o_carrier_id;
    short o_ol_cnt;
    short o_all_local;
    ORDER_LINE_STRUCT o_ol[15];
} ORDERS_STRUCT;

typedef struct

```

```

long c_id;
short c_d_id;
short c_w_id;
char c_first[FIRST_NAME_LEN+1];
char c_middle[MIDDLE_NAME_LEN+1];
char c_last[LAST_NAME_LEN+1];
char c_street_1[ADDRESS_LEN+1];
char c_street_2[ADDRESS_LEN+1];
char c_city[ADDRESS_LEN+1];
char c_state[STATE_LEN+1];
char c_zip[ZIP_LEN+1];
char c_phone[PHONE_LEN+1];
char c_credit[CREDIT_LEN+1];
double c_credit_lim;
double c_discount;
// fix to avoid ODBC float to numeric conversion
// problem.
// double
char c_balance;
char c_balance[6];
double c_ytd_payment;
short c_payment_cnt;
short c_delivery_cnt;
char c_data[C_DATA_LEN+1];
double h_amount;
char h_data[H_DATA_LEN+1];
} CUSTOMER_STRUCT;

typedef struct
{
    char c_last[LAST_NAME_LEN+1];
    char c_first[FIRST_NAME_LEN+1];
    long c_id;
} CUSTOMER_SORT_STRUCT;

typedef struct
{
    long time_start;
} LOADER_TIME_STRUCT;

// Global variables
char szLastError[300];
HENV henv;
HDBC v_hdbc;
verification // for SQL server version

```

```

HDBC      i_hdbc1;
HDBC      // for ITEM table
HDBC      w_hdbc1;
HDBC      // for WAREHOUSE, DISTRICT, STOCK
HDBC      c_hdbc1;
HDBC      // for CUSTOMER
HDBC      c_hdbc2;
HDBC      // for HISTORY
HDBC      o_hdbc1;
HDBC      // for ORDERS
HDBC      o_hdbc2;
HDBC      // for NEW-ORDER
HDBC      o_hdbc3;
HDBC      // for ORDER-LINE

HSTMT     v_hstmt;
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;

TPCC_LDR_ARGS *aptr, args;

//=====
// Function name: main
//=====

int main(int argc, char **argv)
{
    DWORD           dwThreadID[MAX_MAIN_THREADS];
    HANDLE          hThread[MAX_MAIN_THREADS];
    FILE            *fLoader;
    char            buffer[255];
    int             i;

    for (i=0; i<MAX_MAIN_THREADS; i++)
        hThread[i] = NULL;

    printf("\n*****\n");
    printf("\n");
    printf("\n* Microsoft SQL Server
    printf("\n");

```

```

    *");
    printf("\n*
    loader printf("\n* TPC-C BENCHMARK KIT: Database
    *");
    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];
    char        err_log_path[256];

    // 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("idxitm1");

    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);
    strcpy(err_log_path, aptr->log_path);
    strcat(err_log_path, "item.err");
    rc = bcp_init(i_hdbc1, name, NULL,
err_log_path, 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);
            i_name);
            MakeAlphaString(14, 24, I_NAME_LEN,
RandomNumber(100L, 10000L))/100.0;
            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("idxitm1");
    }
}

```

```

//=====
//
// 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];
    char err_log_path[256];

    // Seed with unique number
    seed(2);

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

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
        BuildIndex("idxwarc1");

    InitString(w_name, W_NAME_LEN+1);
    InitAddress(w_street_1, w_street_2, w_city,
w_state, w_zip);

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

    //rc = bcp_init(w_hdbc1, name, NULL,
"logs\warehouse.err", DB_IN);
    strcpy(err_log_path, aptr->log_path);
    strcat(err_log_path, "warehouse.err");
    rc = bcp_init(w_hdbc1, name, NULL,
err_log_path, DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order
(w_id), ROWS_PER_BATCH = %d", aptr->num_warehouses);
        rc = bcp_control(w_hdbc1, BCPHINTS,
(void*) bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);
    }

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

```

```

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

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

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

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

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

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

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

        rc = bcp_bind(w_hdbc1, (BYTE *) &w_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

        time_start = (TimeNow()) / MILLI;
        warehouse_rows_loaded = 0;

        for (w_id = (short)aptr->starting_warehouse;
w_id <= aptr->num_warehouses; w_id++)
        {
            MakeAlphaString(6,10, W_NAME_LEN,
w_name);

            MakeAddress(w_street_1, w_street_2,
w_city, w_state, w_zip);

            w_tax = ((float)
RandomNumber(0L,2000L))/10000.00;

            w_ytd = 300000.00;

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

            warehouse_rows_loaded++;
            CheckForCommit(w_hdbc1, i_hstmt1,
warehouse_rows_loaded, "warehouse", &time_start);
        }

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

```

```

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

        // if build index after load...
        if ((aptr->build_index == 1) && (aptr->index_order == 0))
            BuildIndex("idxwarc1");

        stock_rows_loaded = 0;
        district_rows_loaded = 0;

        District();
        Stock();
    }

//=====
//
// Function : District
//=====
void District()
{
    short d_id;
    short d_w_id;
    char d_name[D_NAME_LEN+1];
    char d_street_1[ADDRESS_LEN+1];
    char d_street_2[ADDRESS_LEN+1];
    char d_city[ADDRESS_LEN+1];
    char d_state[STATE_LEN+1];
    char d_zip[ZIP_LEN+1];
    double d_tax;
    double d_ytd;
    long char name[20];
    long d_next_o_id;
    long time_start;
    int w_id;
    RETCODE rc;
    DBINT rcint;
    char bcphint[128];
    char err_log_path[256];

    // 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);
    strcpy(err_log_path, aptr->log_path);
    strcat(err_log_path, "district.err");
    rc = bcp_init(w_hdbc1, name, NULL,
err_log_path, DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);

```

```

        if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        {
            sprintf(bcphint, "tablock, order
(d_w_id, d_id), ROWS_PER_BATCH = %u", (aptr-
>num_warehouses * 10));
            rc = bcp_control(w_hdbc1, BCPHINTS,
(void*) bcphint);
            if (rc != SUCCEEDED)
                HandleErrorDBC(w_hdbc1);
        }

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

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

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

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

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

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

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

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

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

        rc = bcp_bind(w_hdbc1, (BYTE *) &d_ytd, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 10);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);

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

        d_ytd = 30000.0;
        d_next_o_id = orders_per_district+1;
        time_start = (TimeNow() / MLLI);

```

```

        for (w_id = aptr->starting_warehouse; w_id <=
aptr->num_warehouses; w_id++)
        {
            d_w_id = w_id;

            for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
            {
                MakeAlphastring(6,10,D_NAME_LEN, d_name);

                MakeAddress(d_street_1,
d_street_2, d_city, d_state, d_zip);

                d_tax = ((float)
RandomNumber(0L,2000L))/10000.00;

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

                district_rows_loaded++;
                CheckForCommit(w_hdbc1,
w_hstmt1, district_rows_loaded, "district",
&time_start);
            }

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

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

            // if build index after load...
            if ((aptr->build_index == 1) && (aptr-
>index_order == 0))
                BuildIndex("idxdisc1");

            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];
    char err_log_path[256];

    // Seed with unique number
    seed(3);

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        BuildIndex("idxstk1");

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

    // rc = bcp_init(w_hdbc1, name, NULL,
"logs\\stock.err", DB_IN);
    strcpy(err_log_path, aptr->log_path);
    strcat(err_log_path, "stock.err");
    rc = bcp_init(w_hdbc1, name, NULL,
err_log_path, DB_IN);
    if (rc != SUCCEEDED)
        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 * 10000));
        rc = bcp_control(w_hdbc1, BCPHINTS,
(void*) bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);
    }

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

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

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

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

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

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

```



```

rc = bcp_bind(w_hdbc1, (BYTE *) s_dist_04, 0,
S_DIST_LEN, NULL, 0, 0, 7);
if (rc != 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("idxstk1");

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];
int
num_procs;

char
err_log_path_cust[256];
char
err_log_path_hist[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("idxcust1");
    // check the number of processors
on this system // if 8 or more processors, then
build index on history. // if less than 8 processors, do
not build the index
num_procs = atoi(getenv(
"NUMBER_OF_PROCESSORS"));
if ( num_procs >= 8 )
    BuildIndex("idxhisc1");
}

// Initialize bulk copy
sprintf(name, "%s.%s", aptr->database,
"customer");

//rc = bcp_init(c_hdbc1, name, NULL,
"logs\\customer.err", DB_IN);
strcpy(err_log_path_cust,aptr->log_path);
strcat(err_log_path_cust,"customer.err");
rc = bcp_init(c_hdbc1, name, NULL,
err_log_path_cust, 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));

```

```

(void*) bcphint); rc = bcp_control(c_hdbc1, BCPHINTS,
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc1);
}
sprintf(name, "%.%.%", aptr->database,
"history");
rc = bcp_init(c_hdbc2, name, NULL,
"logs\\history.err", DB_IN);
strcpy(err_log_path_hist, aptr->log_path);
strcat(err_log_path_hist, "history.err");
rc = bcp_init(c_hdbc2, name, NULL,
err_log_path_hist, DB_IN);
if (rc != SUCCEEDED)
    HandleErrorDBC(c_hdbc2);
sprintf(bcp_hint, "tablock");
rc = bcp_control(c_hdbc2, BCPHINTS, (void*)
bcp_hint); if (rc != SUCCEEDED)
    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
        // 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)
    {

```

```

failed in creating creating thread = 0.\n");
        }
        // 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
        {
            printf("Error,
failed in closing customer thread handle with errno:
%d\n", GetLastError());
        }
        if
        {
            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");
            // check the number of processors
            on this system // if 8 or more processors, then
            build index on History. // if less than 8 processors, do
            not build the index // if less than 8 processors, do
            num_procs = atoi(getenv(
"NUMBER_OF_PROCESSORS"));
            if (num_procs >= 8)
                BuildIndex("idxhisc1");
        }
        // 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, "osql -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",
        sprintf(cmd, "osql -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" >
%snurand_load.log",
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database,
        LOADER_NURAND_C,
        aptr-
>log_path);
        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[CREREDIT_LEN+1];
    double c_credit_lim;
    double c_discount;

    // fix to avoid ODBC float to numeric
conversion problem.
    // double c_balance;
    char c_balance[6];

    double c_ytd_payment;
    short c_payment_cnt;
    short c_delivery_cnt;
    char c_data[C_DATA_LEN+1];
    char c_since[C_SINCE_LEN+1];

    RETCODE rc;

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

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

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

    rc = bcp_bind(c_hdbc1, (BYTE *) c_first, 0,
FIRST_NAME_LEN, NULL, 0, 0, 4);
    if (rc != SUCCEEDED)
}
}

```

```

        HandleErrorDBC(c_hdbc1);

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

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

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

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

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

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

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

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

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

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

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

        rc = bcp_bind(c_hdbc1, (BYTE *) &c_discount, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 16);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc1);

        // fix to avoid ODBC float to numeric
        // conversion problem.
        // rc = bcp_bind(c_hdbc1, (BYTE *) &c_balance, 0,
SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 17);
        // if (rc != SUCCEEDED)
        //     HandleErrorDBC(c_hdbc1);

        rc = bcp_bind(c_hdbc1, (BYTE *) c_balance, 0, 5,
NULL, 0, SQLCHARACTER, 17);
        if (rc != SUCCEEDED)

```

```

        HandleErrorDBC(c_hdbc1);

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

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

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

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

        for (i = 0; i < customers_per_district; i++)
        {
            c_id = customer_buf[i].c_id;
            c_d_id = customer_buf[i].c_d_id;
            c_w_id = customer_buf[i].c_w_id;

            strcpy(c_first,
customer_buf[i].c_first);
            strcpy(c_middle,
customer_buf[i].c_middle);
            strcpy(c_last,
customer_buf[i].c_last);
            strcpy(c_street_1,
customer_buf[i].c_street_1);
            strcpy(c_street_2,
customer_buf[i].c_street_2);
            strcpy(c_city,
customer_buf[i].c_city);
            strcpy(c_state,
customer_buf[i].c_state);
            strcpy(c_zip,
customer_buf[i].c_zip);
            strcpy(c_phone,
customer_buf[i].c_phone);
            strcpy(c_credit,
customer_buf[i].c_credit);

            FormatDate(&c_since);

            c_credit_lim =
customer_buf[i].c_credit_lim;
            c_discount =
customer_buf[i].c_discount;

            // fix to avoid ODBC float to
            // numeric conversion problem.
            // c_balance =
customer_buf[i].c_balance;
            strcpy(c_balance,
customer_buf[i].c_balance);

            c_ytd_payment =
customer_buf[i].c_ytd_payment;
            c_payment_cnt =
customer_buf[i].c_payment_cnt;
            c_delivery_cnt =
customer_buf[i].c_delivery_cnt;

```

```

            strcpy(c_data,
customer_buf[i].c_data);

            // Send data to server
            rc = bcp_sendrow(c_hdbc1);
            if (rc != SUCCEEDED)
                HandleErrorDBC(c_hdbc1);

            customer_rows_loaded++;
            CheckForCommit(c_hdbc1, c_hstmt1,
customer_rows_loaded, "customer", &customer_time_start-
>time_start);
        }
    }

//=====
// Function : LoadHistoryTable
//=====

void LoadHistoryTable(LOADER_TIME_STRUCT
*history_time_start)
{
    int i;

    long c_id;
    short c_d_id;
    short c_w_id;
    double h_amount;
    char h_data[H_DATA_LEN+1];
    char h_date[H_DATE_LEN+1];

    RETCODE rc;

    rc = bcp_bind(c_hdbc2, (BYTE *) &c_id, 0,
SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);

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

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

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

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

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

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

```

```

        HandleErrorDBC(c_hdbc2);
    rc = bcp_bind(c_hdbc2, (BYTE *) h_data, 0,
H_DATA_LEN, NULL, 0, 0, 8);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc2);
    for (i = 0; i < customers_per_district; i++)
    {
        c_id = customer_buf[i].c_id;
        c_d_id = customer_buf[i].c_d_id;
        c_w_id = customer_buf[i].c_w_id;
        h_amount =
customer_buf[i].h_amount;
        strcpy(h_data,
customer_buf[i].h_data);

        FormatDate(&h_date);

        // send to server
        rc = bcp_sendrow(c_hdbc2);
        if (rc != SUCCEEDED)
            HandleErrorDBC(c_hdbc2);

        history_rows_loaded++;
        CheckForCommit(c_hdbc2, c_hstmt2,
history_rows_loaded, "history", &history_time_start-
>time_start);
    }
}

//=====
//
// Function : LoadOrders
//
//=====

void LoadOrders()
{
    LOADER_TIME_STRUCT orders_time_start;
    LOADER_TIME_STRUCT new_order_time_start;
    LOADER_TIME_STRUCT
order_line_time_start;
    short d_id; w_id;
    short
    DWORD dwThreadID[MAX_ORDER_THREADS];
    HANDLE
hThread[MAX_ORDER_THREADS];
    char name[20];
    RETCODE rc;
    char bcphint[128];
    char err_log_path_ord[256];
    char err_log_path_nord[256];
    char err_log_path_ord1[256];
    // 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("idxordc1");
            BuildIndex("idxnodc1");
            BuildIndex("idxodc1");
        }
        // initialize bulk copy
        sprintf(name, "%s.%s", aptr->database,
"orders");
        rc = bcp_init(o_hdbc1, name, NULL,
"log\orders.err", DB_IN);
        strcpy(err_log_path_ord, aptr->log_path);
        strcat(err_log_path_ord, "orders.err");
        rc = bcp_init(o_hdbc1, name, NULL,
err_log_path_ord, DB_IN);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc1);

        if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        {
            sprintf(bcphint, "tablock, order
(o_w_id, o_d_id, o_id), ROWS_PER_BATCH = %u", (aptr-
>num_warehouses * 30000));
            rc = bcp_control(o_hdbc1, BCPHINTS,
(void*) bcphint);
            if (rc != SUCCEEDED)
                HandleErrorDBC(o_hdbc1);
        }

        sprintf(name, "%s.%s", aptr->database,
"new_order");
        rc = bcp_init(o_hdbc2, name, NULL,
"log\neword.err", DB_IN);
        strcpy(err_log_path_nord, aptr->log_path);
        strcat(err_log_path_nord, "neword.err");
        rc = bcp_init(o_hdbc2, name, NULL,
err_log_path_nord, DB_IN);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc2);

        if ((aptr->build_index == 1) && (aptr-
>index_order == 1))
        {
            sprintf(bcphint, "tablock, order
(no_w_id, no_d_id, no_o_id), ROWS_PER_BATCH = %u",
(aptr->num_warehouses * 9000));
            rc = bcp_control(o_hdbc2, BCPHINTS,
(void*) bcphint);
            if (rc != SUCCEEDED)
                HandleErrorDBC(o_hdbc2);
        }

        sprintf(name, "%s.%s", aptr->database,
"order_line");
        rc = bcp_init(o_hdbc3, name, NULL,
"log\ordline.err", DB_IN);
        strcpy(err_log_path_ord1, aptr->log_path);
        strcat(err_log_path_ord1, "ordline.err");
        rc = bcp_init(o_hdbc3, name, NULL,
err_log_path_ord1, DB_IN);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc3);

```

```

        if ((aptr->build_index == 1) && (aptr->index_order
== 1))
        {
            sprintf(bcphint, "tablock, order
(o_l_w_id, o_l_d_id, o_l_o_id, o_l_number), ROWS_PER_BATCH
= %u", (aptr->num_warehouses * 30000));
            rc = bcp_control(o_hdbc3, BCPHINTS,
(void*) bcphint);
            if (rc != SUCCEEDED)
                HandleErrorDBC(o_hdbc3);
        }
        orders_rows_loaded = 0;
        new_order_rows_loaded = 0;
        order_line_rows_loaded = 0;
        ordersBufInit();
        orders_time_start.time_start = (TimeNow() /
MILLI);
        new_order_time_start.time_start = (TimeNow()
/ MILLI);
        order_line_time_start.time_start = (TimeNow()
/ MILLI);
        for (w_id = (short)aptr->starting_warehouse;
w_id <= aptr->num_warehouses; w_id++)
        {
            for (d_id = 1; d_id <=
DISTRICT_PER_WAREHOUSE; d_id++)
            {
                OrdersBufLoad(d_id,
w_id);
                // start parallel loading
                threads here...
                // start orders table
                thread
                printf("...Loading Order
Table for: d_id = %d, w_id = %d\n", d_id, w_id);
                CreateThread(NULL,
                    hThread[0] =
                    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
        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].o1
= 0;
            orders_buf[i].o_ol[j].o1_i_id = 0;
            orders_buf[i].o_ol[j].o1_supply_w_id = 0;
            orders_buf[i].o_ol[j].o1_quantity = 0;
            orders_buf[i].o_ol[j].o1_amount = 0;
            strcpy(orders_buf[i].o_ol[j].o1_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 o1;
    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++)
    {
        data // Generate ORDER and NEW-ORDER
            orders_buf[o_id].o_d_id = d_id;
            orders_buf[o_id].o_w_id = w_id;
            orders_buf[o_id].o_id = o_id+1;
            orders_buf[o_id].o_c_id =
cust[o_id+1];
            orders_buf[o_id].o_ol_cnt =
(short)RandomNumber(5L, 15L);
            if (o_id < first_new_order)
            {
                orders_buf[o_id].o_carrier_id =
(short)RandomNumber(1L, 10L);
                orders_buf[o_id].o_all_local = 1;
            }
            else
            {
                orders_buf[o_id].o_carrier_id = 0;
                orders_buf[o_id].o_all_local = 1;
            }
            for (o1=0;
o1<orders_buf[o_id].o_ol_cnt; o1++)
            {
                orders_buf[o_id].o_ol[o1].o1 = o1+1;
                orders_buf[o_id].o_ol[o1].o1_i_id =
RandomNumber(1L, max_items);
                orders_buf[o_id].o_ol[o1].o1_supply_w_id =
w_id;
                orders_buf[o_id].o_ol[o1].o1_quantity = 5;
                MakeAlphaString(24, 24,
OL_DIST_INFO_LEN,
&orders_buf[o_id].o_ol[o1].o1_dist_info);
            }
        }
}

```



```

        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 == aprtr->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 ((captr->build_index == 1) &&
(captr->index_order == 0))
            BuildIndex("idxnodc1");
    }
}

//=====
//
// Function : LoadOrderLineTable
//=====

void LoadOrderLineTable(LOADER_TIME_STRUCT
*order_line_time_start)
{
    long int i,j;
    short 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,
SQL_CHARACTER, 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,
&order_line_time_start->time_start);
        }
    }
    // rcint = bcp_batch(o_hdbc3);
    // if (rcint < 0)
    //     HandleErrorDBC(o_hdbc3);

    if ((o_w_id == aprtr->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 ((captr->build_index == 1) &&
(captr->index_order == 0))
            BuildIndex("idxodl1");
    }
}

//=====
//
// 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,
rows_loaded,          int
*table_name,          char
*time_start)          long
{
    long time_end, time_diff;
    // DBINT rcint;
    if ( !(rows_loaded % apr->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",
        apr->batch,
        table_name,
        time_diff,
        rows_loaded, (float) apr-
>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,
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" ,
    apr->server,
    apr->user,
    apr->password,
    apr->database );

rc = SQLSetConnectOption (i_hdbc1,
SQL_PACKET_SIZE, apr->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" ,
    apr->server,

```

```

    apr->user,
    apr->password,
    apr->database );

rc = SQLSetConnectOption (w_hdbc1,
SQL_PACKET_SIZE, apr->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" ,
    apr->server,
    apr->user,
    apr->password,
    apr->database );

rc = SQLSetConnectOption (c_hdbc1,
SQL_PACKET_SIZE, apr->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" ,
    apr->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, "osql -S%s -U%s -P%s -e -
i%s\\%s.sql > %s%s.log",
    aptr->server,
    aptr->user,
    aptr->password,
    aptr-
>index_script_path,
    index_script,
    aptr->log_path,
    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];
    char err_log_path[256];
    FILE *fp1;

    i = 1;
    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_DBC ,
hdbc1, i, SqlState, &NativeError,
sizeof(Msg) , &MsgLen )) != SQL_NO_DATA )
    {
        printf( szLastError , "%s" , Msg
);
        _strtime(timebuf);
        _strdate(datebuf);
        printf( "[%s : %s] %s\n" , datebuf,
timebuf, szLastError);
        strcpy(err_log_path,aptr-
>log_path);
        strcat(err_log_path,"tpccldr.err");
        fp1 = fopen(err_log_path,"w");
        //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];
    char err_log_path[256];
    FILE *fp1;

    i = 1;
    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_STMT
, hstmt1, i, SqlState, &NativeError,
sizeof(Msg), &MsgLen )) != SQL_NO_DATA )
    {
        sprintf( szLastError, "%s", Msg
);
        _strtime(timebuf);
        _strdate(datebuf);
        printf( "[%s : %s] %s\n", datebuf,
timebuf, szLastError);
        strcpy(err_log_path,aptr-
>log_path);
        strcat(err_log_path,"tpccldr.err");
        fp1 = fopen(err_log_path,"w");
        //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* szTimeOutput )
{
    struct tm when;
    time_t now;

    time( &now );
    when = *localtime( &now );

    mktime( &when );

    // odbc datetime format
    strftime( szTimeOutput, 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]
= {"0000000000"};
    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;

// iterate 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: break;
        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;
}

getargs.c
// File: GETARGS.C
// Microsoft TPC-C
// Kit Ver. 4.41
// Copyright
// Microsoft, 1996, 1997, 1998, 1999, 2000, 2001
// 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("[%d]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->log_path     = LOG_PATH;
    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 '?': /* Fall through */
                GetArgsLoaderUsage();

```

```

                break;
            case 'd':
                pargs->database
                break;
            case 'p':
                pargs->password
                break;
            case 's':
                pargs->server =
                break;
            case 'u':
                pargs->user =
                break;
            case 'b':
                pargs->batch =
                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;
                pargs->log_path
                break;
            case 'l':
                pargs->log_path
                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("[%d]DBG: Entering GetArgsLoaderUsage()\n",
(int) GetCurrentThreadId());
#endif
    printf("TPCCLDR:\n\n");

```



```

}

//=====
// Function : RandomNumber
//
// Description:
//=====
long RandomNumber(long lower, long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%d]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("[%d]DBG: RandomNumber between %ld & %ld
=> %ld\n",
           (int)
GetCurrentThreadId(), lower, upper, rand_num);
#endif

    return rand_num;
}

#if 0
//Original code pgd 08/13/96
long RandomNumber(long lower,
                  long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%d]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("[%d]DBG: RandomNumber between %ld & %ld
=> %ld\n",
           (int)
GetCurrentThreadId(), lower, upper, rand_num);
#endif
}

```

```

#endif

    return rand_num;
}
#endif

//=====
// Function : NURand
//
// Description:
//=====
long NURand(int iConst,
            long x,
            long y,
            long C)
{
    long rand_num;

#ifdef DEBUG
    printf("[%d]DBG: Entering NURand()...\n", (int)
GetCurrentThreadId());
#endif

    rand_num = (((RandomNumber(0,iConst) |
RandomNumber(x,y) + C) % (y-x+1))+x);

#ifdef DEBUG
    printf("[%d]DBG: NURand: num = %d\n", (int)
GetCurrentThreadId(), rand_num);
#endif

    return rand_num;
}

```

strings.c

```

// File: STRINGS.C Microsoft TPC-C
// Kit Ver. 4.41 Copyright
// Microsoft, 1996, 1997, 1998, 1999, 2000, 2001
// 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("[%d]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("[%d]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("[%d]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("[%d]DBG: LastName: num = [%d] ==>
[%d][%d][%d]\n",
           (int)
GetCurrentThreadId(), num, num/100, (num/10)%10,
num%10);
    printf("[%d]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[] =
"0123456789ABCDEFGHIJKLMNPQRSTUWXYZabcdefghijklmnopqrstuvwxyz";
    static int chArrayMax = 61;

#ifdef DEBUG
    printf("[%d]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("[%d]DBG: Entering
MakeOriginalAlphaString()\n", (int)
GetCurrentThreadId());
#endif

    // verify percentage is valid
    if ((percent < 0) || (percent > 100))
    {
        printf("MakeOriginalAlphaString:
Invalid percentage: %d\n", percent);
        exit(-1);
    }

    // verify string is at least 8 chars in length
    if ((x + y) <= 8)
    {
        printf("MakeOriginalAlphaString:
string length must be >= 8\n");
        exit(-1);
    }

    // Make Alpha String
    len = MakeAlphaString(x,y, z, str);

    val = RandomNumber(1,100);
    if (val <= percent)
    {
        start = RandomNumber(0, len - 8);
        strncpy(str + start, "ORIGINAL",
8);
    }

#ifdef DEBUG
    printf("[%d]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, 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, string)

    strcpy(str, "00001111");

    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("[%d]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. 4.41
// Copyright
// Microsoft, 1996, 1997, 1998, 1999, 2000, 2001
// 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 e1_time;

#ifdef DEBUG
    printf("[%d]DBG: Entering TimeNow()\n", (int)
GetCurrentThreadId());
#endif

    _ftime(&e1_time);

    time_now = ((e1_time.time - start_sec) * 1000) +
e1_time.millitm;

    return time_now;
}

```

Appendix C: Tunable Parameters

Microsoft SQL Server 2000 Startup Parameters

Microsoft SQL Server was started with the following command-line options:

```
"C:\Program Files\Microsoft SQL Server\MSSQL\Binn\sqlservr.exe" -e"C:\Program Files\Microsoft SQL Server\MSSQL\LOG\ERRORLOG" -c -x -t3502 -g100
```

Boot.ini

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows Server 2003, Enterprise" /fastdetect /PAE
```

Microsoft Windows Server 2003 EE Configuration (Enabled Services)

These Windows services are started:

- COM+ Event System
- Event Log
- Logical Disk Manager
- Network Connections
- Network Location Awareness (NLA)
- Plug and Play
- Remote Procedure Call (RPC)
- Security Accounts Manager
- System Event Notification
- Terminal Services
- Windows Management Instrumentation

Microsoft SQL Server 2000 Configuration Parameters

name	minimum	maximum
config_value	run_value	

name	minimum	maximum
affinity mask	-2147483648	
2147483647	3	
allow updates	0	1
0	0	
awe enabled	0	1
1	1	
c2 audit mode	0	1
0	0	
cost threshold for parallelism	0	32767
5	5	
Cross DB Ownership Chaining	0	1
0	0	
cursor threshold	-1	-1
2147483647	-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)	0	704
2147483647	0	
lightweight pooling	0	1
1	1	
locks	0	5000
2147483647	0	
max degree of parallelism	0	32
0	0	
max server memory (MB)	4	
2147483647	2147483647	
max text repl size (B)	0	
2147483647	65536	65536
max worker threads	32	32767
255	255	
media retention	0	365
0	0	
min memory per query (KB)	512	
2147483647	1024	1024
min server memory (MB)	0	
2147483647	0	
nested triggers	0	1
1	1	
network packet size (B)	512	65536
4096	4096	
open objects	0	
2147483647	0	
priority boost	0	1
1	1	
query governor cost limit	0	
2147483647	0	
query wait (s)	-1	-1
2147483647	-1	
recovery interval (min)	0	32767
115	115	
remote access	0	1
1	1	
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	

Disk Controller Driver Registry Parameters

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ac2w2k]
"Group"="SCSI miniport"
"Start"=dword:00000000
"Tag"=dword:00000020
"Type"=dword:00000001
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):73,00,79,00,73,00,74,00,74,00,65,00,6d,00,33,00,32,00,5c,00,44,00,\
52,00,49,00,56,00,45,00,52,00,53,00,5c,00,64,00,61,00,63,00,32,00,77,00,32,\
00,6b,00,2e,00,73,00,79,00,73,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ac2w2k\Parameters]
"BusType"=dword:00000008
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ac2w2k\Parameters\Device]
"DriverParameter"="ConfigureSIR=16"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ac2w2k\Parameters\PnpInterface]
"2"=dword:00000001
"5"=dword:00000001
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ac2w2k\Enum]
"0"="PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&254dad54&0&4040"
"Count"=dword:00000003
"NextInstance"=dword:00000003
"1"="PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&94a037d&0&4048"
"2"="PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\4&c59aba9&0&4040"
```

System Summary

System Information report written at: 06/20/05 11:51:44
System Name: SQL3040
[System Summary]

Item	Value
OS Name	Microsoft(R) windows(R) Server 2003, Enterprise Edition
Version	5.2.3790 Build 3790

OS Manufacturer Microsoft Corporation
 Activation Status Activation Pending (36 days remaining)
 System Name SQL3040
 System Manufacturer Itautec
 System Model MX200
 System Type x86-based PC
 Processor x86 Family 15 Model 4 Stepping 3 GenuineIntel ~3400 Mhz
 Processor x86 Family 15 Model 4 Stepping 3 GenuineIntel ~3400 Mhz
 BIOS Version/Date Phoenix Technologies LTD 6.00, 21/1/2005
 SMBIOS Version 2.33
 Windows Directory C:\WINDOWS
 System Directory C:\WINDOWS\system32
 Boot Device \Device\HarddiskVolume4
 Locale Brazil
 Hardware Abstraction Layer Version = "5.2.3790.0 (srv03_rtm.030324-2048)"
 User Name SQL3040\Administrator
 Time Zone E. South America Standard Time
 Total Physical Memory 8.192,00 MB
 Available Physical Memory 191,68 MB
 Total Virtual Memory 17,81 GB
 Available Virtual Memory 2,36 GB
 Page File Space 9,81 GB
 Page File C:\pagefile.sys

[Hardware Resources]

[Conflicts/Sharing]

Resource Device
 I/O Port 0x00000000-0x00000CF7 PCI bus
 I/O Port 0x00000000-0x00000CF7 Direct memory access controller
 Memory Address 0xC4100000-0xCDFFFFFFFF Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596
 Memory Address 0xC4100000-0xCDFFFFFFFF Intel(R) 6700/6702PXH I/OxAPIC Interrupt Controller A - 0326
 I/O Port 0x00002000-0x00006FFF Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596
 I/O Port 0x00002000-0x00006FFF Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329
 I/O Port 0x00002000-0x00006FFF Adaptec AIC-7902B - Ultra320 SCSI
 Memory Address 0xD8000000-0xDBFFFFFFF DEC 21154 PCI to PCI bridge
 Memory Address 0xD8000000-0xDBFFFFFFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 I/O Port 0x00006000-0x00006FFF DEC 21154 PCI to PCI bridge
 I/O Port 0x00006000-0x00006FFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 I/O Port 0x00003000-0x00003FFF DEC 21154 PCI to PCI bridge
 I/O Port 0x00003000-0x00003FFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)

Memory Address 0xC4200000-0xC9FFFFFF Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329
 Memory Address 0xC4200000-0xC9FFFFFF Adaptec AIC-7902B - Ultra320 SCSI
 Memory Address 0xC0000000-0xCDFFFFFFFF DEC 21154 PCI to PCI bridge
 Memory Address 0xC0000000-0xCDFFFFFFFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 Memory Address 0xC8000000-0xC9FFFFFF DEC 21154 PCI to PCI bridge
 Memory Address 0xC8000000-0xC9FFFFFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 Memory Address 0xC6000000-0xC7FFFFFFF DEC 21154 PCI to PCI bridge
 Memory Address 0xC6000000-0xC7FFFFFFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 IRQ 16 Intel(R) E7525/E7520/E7320 PCI Express Root Port A0 - 3595
 IRQ 16 Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596
 IRQ 16 Intel(R) E7525/E7520 PCI Express Root Port B0 - 3597
 IRQ 16 Intel(R) E7520 PCI Express Root Port C0 - 3599
 IRQ 16 Intel(R) 82801EB USB Universal Host Controller - 24D2
 IRQ 16 Intel(R) 82801EB USB Universal Host Controller - 24DE
 I/O Port 0x00005000-0x00006FFF Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A
 I/O Port 0x00005000-0x00006FFF Intel(R) PRO/1000 MT Dual Port Server Adapter
 Memory Address 0xA0000-0xBFFFFFF PCI bus
 Memory Address 0xA0000-0xBFFFFFF RAGE XL PCI Family (Microsoft Corporation)
 I/O Port 0x00004000-0x00004FFF DEC 21154 PCI to PCI bridge
 I/O Port 0x00004000-0x00004FFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 Memory Address 0xD0000000-0xDFFFFFFF Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A
 Memory Address 0xD0000000-0xDFFFFFFF DEC 21154 PCI to PCI bridge
 Memory Address 0xD0000000-0xDFFFFFFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 Memory Address 0xCA000000-0xCDFFFFFFFF Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A
 Memory Address 0xCA000000-0xCDFFFFFFFF Intel(R) PRO/1000 MT Dual Port Server Adapter
 Memory Address 0xD4000000-0xDFFFFFFF Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596
 Memory Address 0xD4000000-0xDFFFFFFF Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329
 Memory Address 0xD4000000-0xDFFFFFFF DEC 21154 PCI to PCI bridge

Memory Address 0xD4000000-0xDFFFFFFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated)
 [DMA]
 Resource Device Status
 Channel 4 Direct memory access controller OK
 Channel 1 ECP Printer Port (LPT1) OK
 Channel 2 Standard Floppy disk controller OK
 [Forced Hardware]
 Device PNP Device ID
 [I/O]
 Resource Device Status
 0x00000000-0x00000CF7 PCI bus OK
 0x00000000-0x00000CF7 Direct memory access controller OK
 0x00000000-0x0000FFFF PCI bus OK
 0x00002000-0x00006FFF Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596 OK
 0x00002000-0x00006FFF Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329 OK
 0x00002000-0x00006FFF Adaptec AIC-7902B - Ultra320 SCSI OK
 0x00003000-0x00003FFF DEC 21154 PCI to PCI bridge OK
 0x00003000-0x00003FFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated) OK
 0x00002400-0x000024FF Adaptec AIC-7902B - Ultra320 SCSI OK
 0x00002C00-0x00002CFF Adaptec AIC-7902B - Ultra320 SCSI OK
 0x00002800-0x000028FF Adaptec AIC-7902B - Ultra320 SCSI OK
 0x00004000-0x00004FFF DEC 21154 PCI to PCI bridge OK
 0x00004000-0x00004FFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated) OK
 0x00005000-0x00006FFF Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A OK
 0x00005000-0x00006FFF Intel(R) PRO/1000 MT Dual Port Server Adapter OK
 0x00006000-0x00006FFF DEC 21154 PCI to PCI bridge OK
 0x00006000-0x00006FFF Mylex extremeRAID 2000 Disk Array Controller (Accelerated) OK
 0x00005040-0x0000507F Intel(R) PRO/1000 MT Dual Port Server Adapter #2 OK
 0x00001400-0x0000141F Intel(R) 82801EB USB Universal Host Controller - 24D2 OK
 0x00001420-0x0000143F Intel(R) 82801EB USB Universal Host Controller - 24D4 OK
 0x00001440-0x0000145F Intel(R) 82801EB USB Universal Host Controller - 24D7 OK
 0x00001460-0x0000147F Intel(R) 82801EB USB Universal Host Controller - 24DE OK
 0x00007000-0x000070FF RAGE XL PCI Family (Microsoft Corporation) OK
 0x000003B0-0x000003BB RAGE XL PCI Family (Microsoft Corporation) OK
 0x000003C0-0x000003DF RAGE XL PCI Family (Microsoft Corporation) OK
 0x00000A79-0x00000A79 ISAPNP Read Data Port OK

0x00000279-0x00000279 OK ISAPNP Read Data Port
0x00000274-0x00000277 OK ISAPNP Read Data Port
0x00000010-0x0000001F OK Motherboard resources
0x00000024-0x00000025 OK Motherboard resources
0x00000028-0x00000029 OK Motherboard resources
0x0000002C-0x0000002D OK Motherboard resources
0x0000002E-0x0000002F OK Motherboard resources
0x0000004E-0x0000004F OK Motherboard resources
0x00000030-0x00000031 OK Motherboard resources
0x00000034-0x00000035 OK Motherboard resources
0x00000038-0x00000039 OK Motherboard resources
0x0000003C-0x0000003D OK Motherboard resources
0x00000050-0x00000053 OK Motherboard resources
0x00000072-0x00000077 OK Motherboard resources
0x00000080-0x00000080 OK Motherboard resources
0x00000090-0x0000009F OK Motherboard resources
0x000000A4-0x000000A5 OK Motherboard resources
0x000000A8-0x000000A9 OK Motherboard resources
0x000000AC-0x000000AD OK Motherboard resources
0x000000B0-0x000000B5 OK Motherboard resources
0x000000B8-0x000000B9 OK Motherboard resources
0x000000BC-0x000000BD OK Motherboard resources
0x000004D0-0x000004D1 OK Motherboard resources
0x00000800-0x000008DF OK Motherboard resources
0x00001000-0x0000107F OK Motherboard resources
0x00001180-0x000011BF OK Motherboard resources
0x00000062-0x00000062 OK Motherboard resources
0x00000066-0x00000066 OK Motherboard resources
0x000000E4-0x000000E6 OK Motherboard resources
0x000000CA2-0x000000CA3 OK Motherboard resources
0x000000CA8-0x000000CAF OK Motherboard resources
0x00000900-0x0000090F OK Motherboard resources
0x0000FE00-0x0000FE00 OK Motherboard resources
0x00000081-0x0000008F OK Direct memory access controller
0x000000C0-0x000000DF OK Direct memory access controller
0x000000F0-0x000000FE OK Numeric data processor

0x00000020-0x00000021 OK Programmable interrupt controller
0x000000A0-0x000000A1 OK Programmable interrupt controller
0x00000070-0x00000071 OK System CMOS/real time clock
0x00000061-0x00000061 OK System speaker
0x00000040-0x00000043 OK System timer
0x00000378-0x0000037F OK ECP Printer Port (LPT1)
0x00000778-0x0000077F OK ECP Printer Port (LPT1)
0x00000060-0x00000060 OK Standard 101/102-Key or Microsoft Natural PS/2 keyboard
0x00000064-0x00000064 OK Standard 101/102-Key or Microsoft Natural PS/2 keyboard
0x000003F0-0x000003F5 OK Standard floppy disk controller
0x000003F7-0x000003F7 OK Standard floppy disk controller
0x000014A0-0x000014AF OK Intel(R) 82801EB Ultra ATA Storage Controllers
0x000001F0-0x000001F7 OK Primary IDE Channel
0x000003F6-0x000003F6 OK Primary IDE Channel
0x00000170-0x00000177 OK Secondary IDE Channel
0x00000376-0x00000376 OK Secondary IDE Channel
0x00001100-0x0000111F OK Intel(R) 82801EB SMBus Controller - 24D3
[IRQs]
Resource Device Status
IRQ 9 Microsoft ACPI-Compliant System OK
IRQ 16 Intel(R) E7525/E7520/E7320 PCI Express Root Port A0 - 3595 OK
IRQ 16 Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596 OK
IRQ 16 Intel(R) E7525/E7520 PCI Express Root Port B0 - 3597 OK
IRQ 16 Intel(R) E7520 PCI Express Root Port C0 - 3599 OK
IRQ 16 Intel(R) 82801EB USB Universal Host Controller - 24D2 OK
IRQ 16 Intel(R) 82801EB USB Universal Host Controller - 24D4 OK
IRQ 24 Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated) OK
IRQ 32 Adaptec AIC-7902B - Ultra320 SCSI OK
IRQ 33 Adaptec AIC-7902B - Ultra320 SCSI OK
IRQ 28 Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated) OK
IRQ 48 Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated) OK
IRQ 54 Intel(R) PRO/1000 MT Dual Port Server Adapter #2 OK
IRQ 19 Intel(R) 82801EB USB Universal Host Controller - 24D4 OK
IRQ 18 Intel(R) 82801EB USB Universal Host Controller - 24D7 OK
IRQ 23 Intel(R) 82801EB USB2 Enhanced Host Controller - 24DD OK

IRQ 17 RAGE XL PCI Family (Microsoft Corporation) OK
IRQ 13 Numeric data processor OK
IRQ 8 System CMOS/real time clock OK
IRQ 0 System timer OK
IRQ 1 Standard 101/102-Key or Microsoft Natural PS/2 keyboard OK
IRQ 12 PS/2 Compatible Mouse OK
IRQ 6 Standard Floppy disk controller OK
IRQ 14 Primary IDE Channel OK
IRQ 15 Secondary IDE Channel OK
IRQ 10 Intel(R) 82801EB SMBus Controller - 24D3 OK

[Memory]

Resource	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	RAGE XL PCI Family (Microsoft Corporation)	OK
0xD4000-0xD7FFF	PCI bus	OK
0xD8000-0xDBFFF	PCI bus	OK
0xDC000-0xDFFFF	PCI bus	OK
0xE0000-0xE3FFF	PCI bus	OK
0xC0000000-0xFEBFFFFF	PCI bus	OK
0xC4100000-0xCDFFFFFFFF	Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596	OK
0xC4100000-0xCDFFFFFFFF	Intel(R) 6700/6702PXH I/OAPIC Interrupt Controller A - 0326	OK
0xD4000000-0xDFFFFFFF	Intel(R) E7525/E7520/E7320 PCI Express Root Port A1 - 3596	OK
0xD4000000-0xDFFFFFFF	Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329	OK
0xD4000000-0xDFFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xD4000000-0xDFFFFFFF	Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated)	OK
0xC4200000-0xC9FFFFFFF	Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329	OK
0xC4200000-0xC9FFFFFFF	Adaptec AIC-7902B - Ultra320 SCSI	OK
0xC6000000-0xC7FFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xC6000000-0xC7FFFFFFF	Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated)	OK
0xC4202000-0xC4203FFF	Adaptec AIC-7902B - Ultra320 SCSI	OK
0xC8000000-0xC9FFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xC8000000-0xC9FFFFFFF	Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated)	OK
0xD8000000-0xDBFFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xD8000000-0xDBFFFFFFF	Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated)	OK
0xCA000000-0xCDFFFFFFFF	Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A	OK
0xCA000000-0xCDFFFFFFFF	Intel(R) PRO/1000 MT Dual Port Server Adapter	OK
0xDC000000-0xDFFFFFFF	Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A	OK
0xDC000000-0xDFFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xDC000000-0xDFFFFFFF	Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated)	OK
0xCC000000-0xCDFFFFFFFF	DEC 21154 PCI to PCI bridge	OK
0xCC000000-0xCDFFFFFFFF	Mylex eXtremeRAID 2000 Disk Array Controller (Accelerated)	OK

```

0xCA020000-0xCA03FFFF Intel(R) PRO/1000 MT Dual
Port Server Adapter #2 OK
0x4101000-0x4101FFF Intel(R) 6700PXH I/OxAPIC
Interrupt Controller B - 0327 OK
0x4000000-0x40003FFF Intel(R) 82801EB USB2
Enhanced Host Controller - 24DD OK
0xCF000000-0xCFFFFFFF RAGE XL PCI Family
(Microsoft Corporation) OK
0xCE000000-0xCE000FFF RAGE XL PCI Family
(Microsoft Corporation) OK
0xE0000000-0xEFFFFFFF Motherboard resources
OK
0xFF800000-0xFFFFFFFF Intel(r) 82802 Firmware
Hub Device OK
0xFE800000-0xFE800FFF Intel(r) 82801EB Ultra
ATA Storage Controllers OK

```

[Components]

[Multimedia]

[Audio Codecs]

CODEC	Manufacturer	Description	Version	Size
c:\windows\system32\sl_anet.acm	Sipro Lab Telecom	Sipro Lab Telecom Audio Codec	OK	3,02 KB (84,00 KB (86.016 bytes))
c:\windows\system32\msg711.acm	Microsoft Corporation	Microsoft	OK	10,00 KB (10.240 bytes)
c:\windows\system32\l3codeca.acm	Fraunhofer Institut Integrierte Schaltungen IIS	Fraunhofer IIS MPEG Layer-3 Codec	OK	1,900,0305 KB (284,00 KB (290.816 bytes))
c:\windows\system32\imaadp32.acm	Microsoft Corporation	Microsoft	OK	15,50 KB (15.872 bytes)
c:\windows\system32\msaud32.acm	Microsoft Corporation	Windows Media Audio Codec	OK	8,000,004,487 KB (288,00 KB (294.912 bytes))
c:\windows\system32\msg723.acm	Microsoft Corporation	Microsoft	OK	4,400,011,350 KB (118.784 bytes)
c:\windows\system32\tssoft32.acm	DSP GROUP, INC.	DSP GROUP, INC.	OK	1,019,50 KB (9.728 bytes)
c:\windows\system32\msadp32.acm	Microsoft Corporation	Microsoft	OK	14,50 KB (14.848 bytes)

```

c:\windows\system32\msgsm32.acm Microsoft Corporation
c:\WINDOWS\system32\MSGSM32.ACM 20,50
5.2.3790.0 (srv03_rtm.030324-2048)
KB (20.992 bytes) 25/3/2003 09:00

```

[Video Codecs]

CODEC	Manufacturer	Description	Version	Size
c:\windows\system32\iyuv_32.dll	Microsoft Corporation	Microsoft	OK	45,00 KB (46.080 bytes)
c:\windows\system32\msh261.drv	Microsoft Corporation	Microsoft	OK	16,50 KB (16.896 bytes)
c:\windows\system32\msh261.drv	Microsoft Corporation	Microsoft	OK	4,400,180,00 KB (184.320 bytes)
c:\windows\system32\msyuv.dll	Microsoft Corporation	Microsoft	OK	26,50 KB (27.136 bytes)
c:\windows\system32\msvidc32.dll	Microsoft Corporation	Microsoft	OK	10,50 KB (10.752 bytes)
c:\windows\system32\msvidc32.dll	Microsoft Corporation	Microsoft	OK	8,00 KB (8.192 bytes)

[CD-ROM]

Item	Value
Drive D:	CD-ROM Drive
Description	No
Media Loaded	CD-ROM
Media Type	SONY CD-ROM CDU5212
Name	(Standard CD-ROM drives)
Manufacturer	OK
Status	Not Available
Transfer Rate	0
SCSI Target ID	IDE\CDROMSONY_CD-ROM_CDU5212
PNP Device ID	5YS1\5&6643C9C&0&0.0.0
Driver	c:\windows\system32\drivers\cdrom.sys (5.2.3790.0 (srv03_rtm.030324-2048), 49,50 KB (50.688 bytes), 25/3/2003 09:00)

[Sound Device]

Item Value

[Display]

Item Value
Name RAGE XL PCI Family (Microsoft Corporation)

```

PNP Device ID PCI\VEN_1002&DEV_4752&SUBSYS_80081002&REV_27\4&3A321F38&0&08F0
Adapter Type ATI RAGE XL PCI (B41), ATI Technologies Inc. compatible
Adapter Description RAGE XL PCI Family (Microsoft Corporation)
Adapter RAM 8,00 MB (8.388.608 bytes)
Installed Drivers ati2drad.dll
Driver Version 5.10.3663.6013
INF File atiixpad.inf (ati2mpad section)
Color Planes 1
Color Table Entries 4294967296
Resolution 1024 x 768 x 75 hertz
Bits/Pixel 32
Memory Address 0xCF000000-0xCFFFFFFF
I/O Port 0x00007000-0x000070FF
Memory Address 0xCE000000-0xCE000FFF
IRQ Channel IRQ 17
I/O Port 0x000003B0-0x000003BB
I/O Port 0x000003C0-0x000003DF
Memory Address 0xA0000-0xBF000
Driver c:\windows\system32\drivers\ati2mpad.sys (5.10.3663.6013, 335,38 KB (343.424 bytes), 27/5/2005 08:27)

```

[Infrared]

Item Value

[Input]

[Keyboard]

Item	Value
Description	Standard 101/102-key or Microsoft Natural PS/2 Keyboard
Name	Enhanced (101- or 102-key)
Layout	00010416
PNP Device ID	ACPI\PNP0303\5&18D3520B&0
Number of Function Keys	12
I/O Port	0x00000060-0x00000060
I/O Port	0x00000064-0x00000064
IRQ Channel	IRQ 1
Driver	c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0 (srv03_rtm.030324-2048), 68,50 KB (70.144 bytes), 25/3/2003 09:00)

[Pointing Device]

Item	Value
Hardware Type	PS/2 Compatible Mouse
Number of Buttons	5
Status	OK
PNP Device ID	ACPI\PNP0F13\5&18D3520B&0
Power Management Supported	No
Double Click Threshold	6
Handedness	Right Handed Operation
IRQ Channel	IRQ 12
Driver	c:\windows\system32\drivers\i8042prt.sys (5.2.3790.0 (srv03_rtm.030324-2048), 68,50 KB (70.144 bytes), 25/3/2003 09:00)

[Modem]

Item Value

[Network]

[Adapter]

```

Item      Value
Name      [00000001] RAS Async Adapter
Adapter Type Not Available
Product Type RAS Async Adapter
Installed Yes
PNP Device ID Not Available
Last Reset 17/6/2005 16:42
Index 1
Service Name AsyncMac
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available

Name [00000002] WAN Miniport (L2TP)
Adapter Type Not Available
Product Type WAN Miniport (L2TP)
Installed Yes
PNP Device ID ROOT\MS_L2TPMINIPORT\0000
Last Reset 17/6/2005 16:42
Index 2
Service Name Rasl2tp
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Driver c:\windows\system32\drivers\rasl2tp.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 77,00 KB (78.848
bytes), 25/3/2003 09:00)

Name [00000003] WAN Miniport (PPTP)
Adapter Type Wide Area Network (WAN)
Product Type WAN Miniport (PPTP)
Installed Yes
PNP Device ID ROOT\MS_PPTPMINIPORT\0000
Last Reset 17/6/2005 16:42
Index 3
Service Name Pptpminiport
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 50:50:54:50:30:30
Driver c:\windows\system32\drivers\rasppptp.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 70,50 KB (72.192
bytes), 25/3/2003 09:00)

Name [00000004] WAN Miniport (PPPOE)
Adapter Type Wide Area Network (WAN)
Product Type WAN Miniport (PPPOE)
Installed Yes
PNP Device ID ROOT\MS_PPPOEMINIPORT\0000
Last Reset 17/6/2005 16:42
Index 4
Service Name Raspppoe

```

```

IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 33:50:6F:45:30:30
Driver c:\windows\system32\drivers\raspppoe.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 38,00 KB (38.912
bytes), 25/3/2003 09:00)

Name [00000005] Direct Parallel
Adapter Type Not Available
Product Type Direct Parallel
Installed Yes
PNP Device ID ROOT\MS_PTMINIPORT\0000
Last Reset 17/6/2005 16:42
Index 5
Service Name Raspti
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Driver c:\windows\system32\drivers\raspti.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 18,50 KB (18.944
bytes), 25/3/2003 09:00)

Name [00000006] WAN Miniport (IP)
Adapter Type Not Available
Product Type WAN Miniport (IP)
Installed Yes
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 17/6/2005 16:42
Index 6
Service Name Ndiswan
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available
Driver c:\windows\system32\drivers\ndiswan.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 96,50 KB (98.816
bytes), 25/3/2003 09:00)

Name [00000007] Intel(R) PRO/1000 MT Dual Port
Server Adapter
Adapter Type Ethernet 802.3
Product Type Intel(R) PRO/1000 MT Dual Port
Server Adapter
Installed Yes
PNP Device ID PCI\VEN_8086&DEV_1079&SUBSYS_117A8086&REV_03\
5&2199E002&0&100218
Last Reset 17/6/2005 16:42
Index 7
Service Name E1000
IP Address 192.168.20.54
IP Subnet 255.255.255.0
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:30:48:2D:69:D4

```

```

Memory Address 0xCA000000-0xCDFFFFFFFF
I/O Port 0x00005000-0x00006FFF
IRQ Channel IRQ 54
Driver c:\windows\system32\drivers\e1000325.sys
(8.4.21.0 built by: winddk, 172,00 KB (176.128 bytes),
30/5/2005 14:09)

```

```

Name [00000008] Intel(R) PRO/1000 MT Dual Port
Server Adapter
Adapter Type Ethernet 802.3
Product Type Intel(R) PRO/1000 MT Dual Port
Server Adapter
Installed Yes
PNP Device ID PCI\VEN_8086&DEV_1079&SUBSYS_117A8086&REV_03\
5&2199E002&0&110218
Last Reset 17/6/2005 16:42
Index 8
Service Name E1000
IP Address 0.0.0.0
IP Subnet 0.0.0.0
Default IP Gateway Not Available
DHCP Enabled Yes
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:30:48:2D:69:D5
Memory Address 0xCA020000-0xCA03FFFFF
I/O Port 0x00005040-0x0000507F
IRQ Channel IRQ 55
Driver c:\windows\system32\drivers\e1000325.sys
(8.4.21.0 built by: winddk, 172,00 KB (176.128 bytes),
30/5/2005 14:09)

```

[Protocol]

```

Item      Value
Name      MSAFD Tcpip [TCP/IP]
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 16 bytes
Maximum Message Size 0 bytes
Message Oriented No
Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data Yes
Supports Graceful Closing Yes
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name      MSAFD Tcpip [UDP/IP]
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 16 bytes
Maximum Message Size 63,93 KB (65.467 bytes)

Message Oriented Yes
Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No

```


Item Value
 [Parallel]
 Item Value
 Name LPT1
 PNP Device ID ACPI\PNP0401\5&18d3520b&0
 I/O Port 0x00000378-0x0000037F
 I/O Port 0x00000778-0x0000077F
 DMA Channel Channel 1
 Driver c:\windows\system32\drivers\parport.sys
 (5.2.3790.0 (srv03_rtm.030324-2048), 76,50 KB (78.336 bytes), 24/3/2003 20:04)

[Storage]

[Drives]

Item Value
 Drive A:
 Description 3 1/2 Inch Floppy Drive
 Drive C:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 34,22 GB (36.742.291.456 bytes)
 Free Space 29,90 GB (32.108.097.536 bytes)

Volume Name
 Volume Serial Number 9CBE8012

Drive D:
 Description CD-ROM Disc

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 Removable Disk
 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 W:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 634,77 GB (681.579.597.824 bytes)
 Free Space 521,94 GB (560.432.107.520 bytes)

Volume Name tpccback1
 Volume Serial Number ACB17E01

Drive X:
 Description Local Fixed Disk
 Compressed No
 File System NTFS
 Size 634,77 GB (681.579.597.824 bytes)
 Free Space 521,94 GB (560.432.173.056 bytes)

Volume Name tpccback2
 Volume Serial Number 40BC2562

[Disks]

Item Value
 Description Disk drive
 Manufacturer (Standard disk drives)
 Model MAXTOR ATLAS15K_36SCA SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 2
 SCSI Target ID 0
 Sectors/Track 63
 Size 34,22 GB (36.742.325.760 bytes)
 Total Cylinders 4.467
 Total Sectors 71.762.355
 Total Tracks 1.139.085
 Tracks/Cylinder 255
 Partition Disk #0, Partition #0
 Partition Size 34,22 GB (36.742.293.504 bytes)

Partition Starting Offset 32.256 bytes

Description Disk drive
 Manufacturer (Standard disk drives)
 Model MAXTOR ATLAS15K_36SCA SCSI Disk Device
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 1
 SCSI Bus 0
 SCSI Logical Unit 0
 SCSI Port 2
 SCSI Target ID 1
 Sectors/Track 63
 Size 34,22 GB (36.742.325.760 bytes)
 Total Cylinders 4.467

Total Sectors 71.762.355
 Total Tracks 1.139.085
 Tracks/Cylinder 255
 Partition Disk #1, Partition #0
 Partition Size 34,21 GB (36.734.068.224 bytes)

Partition Starting Offset 32.256 bytes

Description \\.\PHYSICALDRIVE4
 Manufacturer Not Available
 Model Not Available
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 3
 SCSI Bus 4

SCSI Logical Unit 0
 SCSI Port 6
 SCSI Target ID 0
 Sectors/Track 63
 Size 956,64 GB (1.027.189.416.960 bytes)
 Total Cylinders 124.882
 Total Sectors 2.006.229.330
 Total Tracks 31.844.910
 Tracks/Cylinder 255
 Partition Disk #4, Partition #0
 Partition Size 58,59 GB (62.915.134.464 bytes)

Partition Starting Offset 32.256 bytes

Partition Disk #4, Partition #1
 Partition Size 107,43 GB (115.351.326.720 bytes)

Partition Starting Offset 62.915.166.720 bytes

Partition Disk #4, Partition #2
 Partition Size 634,77 GB (681.579.601.920 bytes)

Partition Starting Offset 178.266.493.440 bytes

Description \\.\PHYSICALDRIVE2
 Manufacturer Not Available
 Model Not Available
 Bytes/Sector 512
 Media Loaded Yes
 Media Type Fixed hard disk
 Partitions 3
 SCSI Bus 4

SCSI Logical Unit 0
 SCSI Port 4
 SCSI Target ID 0
 Sectors/Track 63
 Size 956,64 GB (1.027.189.416.960 bytes)
 Total Cylinders 124.882
 Total Sectors 2.006.229.330
 Total Tracks 31.844.910
 Tracks/Cylinder 255
 Partition Disk #2, Partition #0
 Partition Size 58,59 GB (62.915.134.464 bytes)

Partition Starting Offset 32.256 bytes

Partition Disk #2, Partition #1
 Partition Size 107,43 GB (115.351.326.720 bytes)

Partition Starting Offset 62.915.166.720 bytes

Partition Disk #2, Partition #2
 Partition Size 634,77 GB (681.579.601.920 bytes)

Partition Starting Offset 178.266.493.440 bytes


```

Description      \\.\PHYSICALDRIVE3
Manufacturer     Not Available
Model           Not Available
Bytes/Sector    512
Media Loaded    Yes
Media Type      Fixed hard disk
Partitions      1
SCSI Bus        4
SCSI Logical Unit 0
SCSI Port       5
SCSI Target ID  0
Sectors/Track  63
Size            136,66 GB (146.738.995.200 bytes)
Total cylinders 17,840
Total Sectors   286,599,600
Total Tracks    4,549,200
Tracks/Cylinder 255
Partition Disk #3, Partition #0
Partition Size  136,66 GB (146.738.962.944 bytes)

Partition Starting Offset 32.256 bytes

```

```

Description      Disk drive
Manufacturer     (Standard disk drives)
Model           Flash Drive OT_USB20 USB Device
Bytes/Sector    512
Media Loaded    Yes
Media Type      Removable media
Partitions      1
SCSI Bus        Not Available
SCSI Logical Unit Not Available
SCSI Port       Not Available
SCSI Target ID  Not Available
Sectors/Track  63
Size            117,66 MB (123.379.200 bytes)
Total cylinders  15
Total Sectors   240,975
Total Tracks    3,825
Tracks/Cylinder 255
Partition Disk #5, Partition #0
Partition Size  124,98 MB (131.055.104 bytes)
Partition Starting Offset 16.384 bytes

```

[SCSI]

```

Item      Value
Name      Mylex extremeRAID 2000 Disk Array Controller
(Accelerated)
Manufacturer Mylex
Status    OK
PNP Device ID
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\
6&19B2D4A0&0&40080018
Memory Address 0xc6000000-0xc7ffffff
I/O Port 0x00003000-0x00003fff
Memory Address 0xd4000000-0xdfffffff
IRQ Channel  IRQ 24
Driver c:\windows\system32\drivers\dac2w2k.sys
(7.00-14, 172,75 KB (176.896 bytes), 23/9/2003 10:46)

```

```

Name      Adaptec AIC-7902B - Ultra320 SCSI
Manufacturer Adaptec
Status    OK
PNP Device ID
PCI\VEN_9005&DEV_801D&SUBSYS_548015D9&REV_10\
5&d4388E4&0&100018
I/O Port 0x00002400-0x000024ff
Memory Address 0xc4200000-0xc9ffffff
I/O Port 0x00002000-0x00006fff
IRQ Channel  IRQ 32

```

```

Driver c:\windows\system32\drivers\adpu320.sys
(3.0.000.000 built by: winddk, 129,50 KB (132.608
bytes), 27/5/2005 08:18)
Name      Adaptec AIC-7902B - Ultra320 SCSI
Manufacturer Adaptec
Status    OK
PNP Device ID
PCI\VEN_9005&DEV_801D&SUBSYS_548015D9&REV_10\
5&d4388E4&0&100018
I/O Port 0x00002c00-0x00002cfff
Memory Address 0xc4202000-0xc4203fff
I/O Port 0x00002800-0x000028ff
IRQ Channel  IRQ 33
Driver c:\windows\system32\drivers\adpu320.sys
(3.0.000.000 built by: winddk, 129,50 KB (132.608
bytes), 27/5/2005 08:18)

```

```

Name      Mylex extremeRAID 2000 Disk Array Controller
(Accelerated)
Manufacturer Mylex
Status    OK
PNP Device ID
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\
6&3057A610&0&40180018
Memory Address 0xc8000000-0xc9ffffff
I/O Port 0x00004000-0x00004fff
Memory Address 0xd8000000-0xdbffffff
IRQ Channel  IRQ 28
Driver c:\windows\system32\drivers\dac2w2k.sys
(7.00-14, 172,75 KB (176.896 bytes), 23/9/2003 10:46)

```

```

Name      Mylex extremeRAID 2000 Disk Array Controller
(Accelerated)
Manufacturer Mylex
Status    OK
PNP Device ID
PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\
6&360400C6&0&40080218
Memory Address 0xc0000000-0xcdffffff
I/O Port 0x00006000-0x00006fff
Memory Address 0xdc000000-0xdfffffff
IRQ Channel  IRQ 48
Driver c:\windows\system32\drivers\dac2w2k.sys
(7.00-14, 172,75 KB (176.896 bytes), 23/9/2003 10:46)

```

[IDE]

```

Item      Value
Name      Intel(R) 82801EB Ultra ATA Storage
Controllers
Manufacturer Intel
Status    OK
PNP Device ID
PCI\VEN_8086&DEV_24DB&SUBSYS_548015D9&REV_02\
3&61AAA01&0&F9
I/O Port 0x000014A0-0x000014AF
Memory Address 0xfefbfcc0-0xfefbffff
Driver c:\windows\system32\drivers\pciide.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 5,50 KB (5.632
bytes), 8/6/2005 18:22)

```

```

Name      Primary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI
controllers)
Status    OK
PNP Device ID PCIIDE\IDECHANNEL\4&7145BEC&0&0
I/O Port 0x000001F0-0x000001F7
I/O Port 0x000003F6-0x000003F6

```

```

IRQ Channel  IRQ 14
Driver c:\windows\system32\drivers\atapi.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 89,00 KB (91.136
bytes), 8/6/2005 18:22)

```

```

Name      Secondary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI
controllers)
Status    OK
PNP Device ID PCIIDE\IDECHANNEL\4&7145BEC&0&1

```

```

I/O Port 0x00000170-0x00000177
I/O Port 0x00000376-0x00000376
IRQ Channel  IRQ 15
Driver c:\windows\system32\drivers\atapi.sys
(5.2.3790.0 (srv03_rtm.030324-2048), 89,00 KB (91.136
bytes), 8/6/2005 18:22)

```

[Printing]

Name	Driver	Port	Server Name

[Problem Devices]

Device	PNP Device ID	Error Code

[USB]

```

Device      PNP Device ID
Intel(R) 82801EB USB Universal Host Controller - 24D2
PCI\VEN_8086&DEV_24D2&SUBSYS_548015D9&REV_02\
3&61AAA01&0&E8
USB Root Hub USB\ROOT_HUB\4&10B6329E&0
Intel(R) 82801EB USB Universal Host Controller - 24D4
PCI\VEN_8086&DEV_24D4&SUBSYS_548015D9&REV_02\
3&61AAA01&0&E9
USB Root Hub USB\ROOT_HUB\4&949A15D&0
Intel(R) 82801EB USB Universal Host Controller - 24D7
PCI\VEN_8086&DEV_24D7&SUBSYS_548015D9&REV_02\
3&61AAA01&0&EA
USB Root Hub USB\ROOT_HUB\4&33788DB3&0
Intel(R) 82801EB USB Universal Host Controller - 24DE
PCI\VEN_8086&DEV_24DE&SUBSYS_548015D9&REV_02\
3&61AAA01&0&EB
USB Root Hub USB\ROOT_HUB\4&353EC4EB&0
Intel(R) 82801EB USB2 Enhanced Host Controller - 24DD
PCI\VEN_8086&DEV_24DD&SUBSYS_548015D9&REV_02\
3&61AAA01&0&EF
USB Root Hub USB\ROOT_HUB20\4&2B85A8E&0
USB Mass Storage Device
USB\VID_0EA0&PID_2168\382F41ACD2950022
Flash Drive OT_USB20 USB Device
USBSTOR\DISK&VEN_FLASH&PROD_DRIVE_OT_USB20&RE
V_2_00\382F41ACD2950022&0
Generic volume
STORAGE\REMOVABLEMEDIA\7&EAF53AF&0&RM

```

[Software Environment]

[System Drivers]

Name	Description	Start Mode	File	Type	State
abiosdsk	Abiosdsk	Not Available		Kernel Driver	
	No	Disabled	Stopped	OK	
	Ignore	No	No		
acpi	Microsoft ACPI Driver		c:\windows\system32\drivers\acpi.sys		

	Kernel Driver	Yes	Boot		
	Running OK	Normal	No	Yes	
acpiec	ACPIEC c:\windows\system32\drivers\acpiec.sys	No	Disabled	Stopped	No
adpu160m	adpu160m	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
adpu320	adpu320	No	Kernel Driver		
	c:\windows\system32\drivers\adpu320.sys	Yes	Boot	Running OK	Yes
afcnt	afcnt	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
afd	AFD Networking Support Environment c:\windows\system32\drivers\afd.sys	Yes	Auto	Running OK	Yes
aha154x	Aha154x	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
aic78u2	aic78u2	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
aic78xx	aic78xx	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
aliide	AliIde	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
asyncmac	RAS Asynchronous Media Driver c:\windows\system32\drivers\asyncmac.sys	No	Manual	Stopped OK	No
atapi	Standard IDE/ESDI Hard Disk Controller c:\windows\system32\drivers\atapi.sys	Yes	Boot	Running OK	Yes
atdisk	Atdisk	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
ati2mpad	ati2mpad	No	Manual	Running OK	Yes
atmarpc	ATM ARP Client Protocol c:\windows\system32\drivers\atmarpc.sys	No	Manual	Stopped OK	No
audstub	Audio Stub Driver c:\windows\system32\drivers\audstub.sys	Yes	Manual	Running OK	Yes
beep	Beep c:\windows\system32\drivers\beep.sys	Yes	System	Running OK	Yes
cbidf2k	cbidf2k	No	Disabled	Running OK	

	Stopped	OK	Normal	No	No
cd20xrnt	cd20xrnt	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
cdfs	Cdfs	No	File	Running OK	
	c:\windows\system32\drivers\cdfs.sys	Yes	System	Running OK	
cdrom	CD-ROM Driver c:\windows\system32\drivers\cdrom.sys	Yes	System	Running OK	Yes
changer	Changer	Not Available	Kernel Driver		
	No	System	Stopped	OK	
clusdisk	Cluster Disk Driver c:\windows\system32\drivers\clusdisk.sys	No	Disabled	Running OK	No
cmdide	CmdIde	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
cpqarray	Cpqarray	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
cpqarray2	Cpqarray2	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
cpqcissm	Cpqcissm	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
cpqfcalm	Cpqfcalm	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
crcdisk	CRC Disk Filter Driver c:\windows\system32\drivers\crcdisk.sys	Yes	Boot	Running OK	Yes
dac2w2k	dac2w2k	No	Boot	Running OK	Yes
	c:\windows\system32\drivers\dac2w2k.sys	Yes	Normal	Running OK	Yes
dac960nt	dac960nt	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
dellcerc	dellcerc	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
dfsdriver	DfsDriver	No	Boot	Running OK	Yes
	c:\windows\system32\drivers\dfs.sys	Yes	Normal	Running OK	Yes
disk	Disk Driver c:\windows\system32\drivers\disk.sys	Yes	Boot	Running OK	Yes
dmbboot	dmbboot	No	Disabled	Running OK	No
	c:\windows\system32\drivers\dmbboot.sys	No	Normal	Running OK	No
dmio	Logical Disk Manager Driver c:\windows\system32\drivers\dmio.sys	Yes	Boot	Running OK	Yes

dmload	dmload				
	c:\windows\system32\drivers\dmload.sys	Yes	Boot	Running OK	Yes
dpti2o	dpti2o	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
e1000	Intel(R) PRO/1000 Network Connection Driver c:\windows\system32\drivers\e1000325.sys	No	Manual	Running OK	Yes
fastfat	Fastfat				
	c:\windows\system32\drivers\fastfat.sys	Yes	Disabled	Running OK	File
fdc	Floppy Disk Controller Driver c:\windows\system32\drivers\fdc.sys	Yes	Manual	Running OK	Yes
fips	Fips				
	c:\windows\system32\drivers\fips.sys	Yes	System	Running OK	Yes
flpydisk	Floppy Disk Driver c:\windows\system32\drivers\flpydisk.sys	Yes	Manual	Running OK	Yes
ftdisk	Volume Manager Driver c:\windows\system32\drivers\ftdisk.sys	Yes	Boot	Running OK	Yes
gpc	Generic Packet Classifier c:\windows\system32\drivers\msgpc.sys	Yes	Manual	Running OK	Yes
hpn	hpn	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
hpt3xx	hpt3xx	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
http	HTTP				
	c:\windows\system32\drivers\http.sys	No	Manual	Running OK	No
i2omgmt	i2omgmt	Not Available	Kernel Driver		
	No	System	Stopped	OK	
i2omp	i2omp	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver c:\windows\system32\drivers\i8042prt.sys	Yes	System	Running OK	Yes
iirsp	iirsp	Not Available	Kernel Driver		
	No	Disabled	Stopped	OK	
imapi	CD-Burning Filter Driver c:\windows\system32\drivers\imapi.sys	No	System	Running OK	No

intelide	IntelIde c:\windows\system32\drivers\intelide.sys Kernel Driver Running OK	Yes Normal	Boot No	Yes
ipfilterdriver	IP Traffic Filter Driver c:\windows\system32\drivers\ipfltdrv.sys Kernel Driver Stopped OK	No Normal	Manual No	No
ipinip	IP in IP Tunnel Driver c:\windows\system32\drivers\ipinip.sys Kernel Driver Stopped OK	No Normal	Manual No	No
ipnat	IP Network Address Translator c:\windows\system32\drivers\ipnat.sys Kernel Driver Stopped OK	No Normal	Manual No	No
ipsec	IPSEC driver c:\windows\system32\drivers\ipsec.sys Kernel Driver Running OK	Yes Normal	System No	Yes
ipsraidn	ipsraidn Not Available No Disabled Stopped OK	No	Kernel Driver	
isapnp	PnP ISA/EISA Bus Driver c:\windows\system32\drivers\isapnp.sys Kernel Driver Running OK	Yes Critical	Boot No	Yes
kbdclass	Keyboard Class Driver c:\windows\system32\drivers\kbdclass.sys Kernel Driver Running OK	Yes Normal	System No	Yes
ksecdd	KSecDD c:\windows\system32\drivers\ksecdd.sys Kernel Driver Running OK	Yes Normal	Boot No	Yes
lp6nds35	lp6nds35 Not Available No Disabled Stopped OK	No	Kernel Driver	
mac2w2k	mac2w2k c:\windows\system32\drivers\mac2w2k.sys Kernel Driver Running OK	Yes Normal	Boot No	Yes
mnmdd	mnmdd c:\windows\system32\drivers\mnmdd.sys Kernel Driver Running OK	Yes Ignore	System No	Yes
modem	Modem c:\windows\system32\drivers\modem.sys Kernel Driver Stopped OK	No Ignore	Manual No	No
mouclass	Mouse Class Driver c:\windows\system32\drivers\mouclass.sys Kernel Driver Running OK	Yes Normal	System No	Yes
mountmgr	Mount Point Manager c:\windows\system32\drivers\mountmgr.sys Kernel Driver Running OK	Yes Normal	Boot No	Yes

mraid35x	mraid35x Not Available No Disabled Stopped OK	No	Kernel Driver	
mrxdav	WebDav Client Redirector c:\windows\system32\drivers\mrxdav.sys System Driver Normal	No No	Manual Stopped OK	File
mrxsmb	MRXSMB c:\windows\system32\drivers\mrxsmb.sys System Driver Normal	Yes No	System Yes	Running File OK
msfs	MSfs c:\windows\system32\drivers\msfs.sys System Driver Normal	Yes No	System Yes	Running File OK
mup	Mup c:\windows\system32\drivers\mup.sys File system Driver Running OK	Yes Normal	System Boot No	Running File Yes
ndis	NDIS System Driver c:\windows\system32\drivers\ndis.sys Kernel Driver Running OK	Yes Normal	Boot No	Yes
ndistapi	Remote Access NDIS TAPI Driver c:\windows\system32\drivers\ndistapi.sys Kernel Driver Running OK	Yes Normal	Manual No	Yes
ndisuio	NDIS Usermode I/O Protocol c:\windows\system32\drivers\ndisuio.sys Kernel Driver Stopped OK	No Normal	Manual No	No
ndiswan	Remote Access NDIS WAN Driver c:\windows\system32\drivers\ndiswan.sys Kernel Driver Running OK	Yes Normal	Manual No	Yes
ndproxy	NDIS Proxy c:\windows\system32\drivers\ndproxy.sys Kernel Driver Running OK	Yes Normal	Manual No	Yes
netbios	NetBIOS Interface c:\windows\system32\drivers\netbios.sys System Driver Normal	Yes No	System Yes	Running File OK
netbt	NetBios over Tcpip c:\windows\system32\drivers\netbt.sys Kernel Driver Running OK	Yes Normal	System No	Yes
nfrd960	nfrd960 Not Available No Disabled Stopped OK	No	Kernel Driver	
npfs	Npfs c:\windows\system32\drivers\npfs.sys System Driver Normal	Yes No	System Yes	Running File OK
ntfs	Ntfs c:\windows\system32\drivers\ntfs.sys System Driver Normal	Yes No	Disabled Yes	Running File OK
null	Null c:\windows\system32\drivers\null.sys Kernel Driver Running OK	Yes Normal	System No	Yes
parport	Parallel port driver c:\windows\system32\drivers\parport.sys Kernel Driver Normal	Yes Manual		

partmgr	Partition Manager c:\windows\system32\drivers\partmgr.sys Kernel Driver Running OK	Yes Normal	Boot No	Yes
parvdm	Parvdm c:\windows\system32\drivers\parvdm.sys Kernel Driver Running OK	Yes Ignore	Auto No	Yes
pci	PCI Bus Driver c:\windows\system32\drivers\pci.sys Kernel Driver Running OK	Yes Critical	Boot No	Yes
pciide	PCIIde c:\windows\system32\drivers\pciide.sys Kernel Driver Running OK	Yes Normal	Boot No	Yes
pcmcia	Pcmcia c:\windows\system32\drivers\pcmcia.sys Kernel Driver Stopped OK	No Normal	Disabled No	No
pdcomp	PDCOMP Not Available No Manual Stopped OK	No	Kernel Driver	
pdframe	PDFRAME Not Available No Manual Stopped OK	No	Kernel Driver	
pdreli	PDRELI Not Available No Manual Stopped OK	No	Kernel Driver	
pdrframe	PDRFRAME Not Available No Manual Stopped OK	No	Kernel Driver	
perc2	perc2 Not Available No Disabled Stopped OK	No	Kernel Driver	
perc2hib	perc2hib Not Available No Disabled Stopped OK	No	Kernel Driver	
pptpminiport	WAN Miniport (PPTP) c:\windows\system32\drivers\rasppptp.sys Kernel Driver Running OK	Yes Normal	Manual No	Yes
processor	Processor Driver c:\windows\system32\drivers\processr.sys Kernel Driver Running OK	Yes Normal	Manual No	Yes
ptilink	Direct Parallel Link Driver c:\windows\system32\drivers\ptilink.sys Kernel Driver Running OK	Yes Normal	Manual No	Yes
q11080	q11080 Not Available No Disabled Stopped OK	No	Kernel Driver	
q110wnt	q110wnt Not Available No Disabled Stopped OK	No	Kernel Driver	
q112160	q112160 Not Available No Disabled Stopped OK	No	Kernel Driver	
q11240	q11240 Not Available No Disabled Stopped OK	No	Kernel Driver	

q11280	q11280	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
q12100	q12100	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
q12200	q12200	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
q12300	q12300	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
rasacd	Remote Access Auto Connection Driver		
	c:\windows\system32\drivers\rasacd.sys	Yes	System
	Kernel Driver	Yes	Running
	Normal	No	Yes
rasl2tp	WAN Miniport (L2TP)		
	c:\windows\system32\drivers\rasl2tp.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
raspppoe	Remote Access PPPoE Driver		
	c:\windows\system32\drivers\raspppoe.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
raspti	Direct Parallel		
	c:\windows\system32\drivers\raspti.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
rdbss	Rdbss		
	c:\windows\system32\drivers\rdbss.sys	File	Running
System Driver	Yes	System	OK
	Normal	No	Yes
rdpcdd	RDPcdd		
	c:\windows\system32\drivers\rdpcdd.sys	Yes	System
	Kernel Driver	Yes	Running
	Normal	Ignore	No
rdpdr	Terminal Server Device Redirector Driver		
	c:\windows\system32\drivers\rdpdr.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
rdpwd	RDPWD		
	c:\windows\system32\drivers\rdpwd.sys	No	Manual
	Kernel Driver	No	Stopped
	Normal	Ignore	No
redbook	Digital CD Audio Playback Filter Driver		
	c:\windows\system32\drivers\redbook.sys	Yes	System
	Kernel Driver	Yes	Running
	Normal	No	Yes
secdrv	Secdrv		
	c:\windows\system32\drivers\secdrv.sys	No	Manual
	Kernel Driver	No	Stopped
	Normal	No	No
serenum	Serenum Filter Driver		
	c:\windows\system32\drivers\serenum.sys	No	Manual
	Kernel Driver	No	Stopped
	Normal	No	No
serial	Serial port driver		
	c:\windows\system32\drivers\serial.sys	No	System
	Kernel Driver	No	Stopped
	Normal	Ignore	No

sfloppy	Sfloppy		
	c:\windows\system32\drivers\sfloppy.sys	No	System
	Kernel Driver	No	Stopped
	Normal	Ignore	No
simbad	Simbad	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
sparrow	Sparrow	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
srv	Srv		
	c:\windows\system32\drivers\srv.sys	No	Manual
	File System Driver	No	Stopped
	Normal	No	No
swenum	Software Bus Driver		
	c:\windows\system32\drivers\swenum.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
symc810	symc810	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
symc8xx	symc8xx	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
symmpi	symmpi	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
sym_hi	sym_hi	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
sym_u3	sym_u3	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
tcpip	TCP/IP Protocol Driver		
	c:\windows\system32\drivers\tcpip.sys	Yes	System
	Kernel Driver	Yes	Running
	Normal	Ignore	No
tdpipe	TDPIPE		
	c:\windows\system32\drivers\tdpipe.sys	No	Manual
	Kernel Driver	No	Stopped
	Normal	Ignore	No
tdtcp	TDTCP		
	c:\windows\system32\drivers\tdtcp.sys	No	Manual
	Kernel Driver	No	Stopped
	Normal	Ignore	No
termdd	Terminal Device Driver		
	c:\windows\system32\drivers\termdd.sys	Yes	System
	Kernel Driver	Yes	Running
	Normal	No	Yes
toside	Toside	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
udfs	Udfs		
	c:\windows\system32\drivers\udfs.sys	File	Running
System Driver	No	Disabled	Stopped
	Normal	No	OK
ultra	ultra	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
update	Microcode Update Driver		
	c:\windows\system32\drivers\update.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
usbhci	Microsoft USB 2.0 Enhanced Host Controller		
	Miniport Driver		

	c:\windows\system32\drivers\usbhci.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
usbhub	USB2 Enabled Hub		
	c:\windows\system32\drivers\usbhub.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
usbstor	USB Mass Storage Driver		
	c:\windows\system32\drivers\usbstor.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
usbuhci	Microsoft USB Universal Host Controller		
	Miniport Driver		
	c:\windows\system32\drivers\usbuhci.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
vgasave	VGA Display Controller		
	c:\windows\system32\drivers\vga.sys	Yes	System
	Kernel Driver	Yes	Running
	Ignore	No	Yes
viaide	ViaIde	Not Available	Kernel Driver
	No	Disabled	Stopped
	Normal	No	OK
volsnap	Storage volumes		
	c:\windows\system32\drivers\volsnap.sys	Yes	Boot
	Kernel Driver	Yes	Running
	Normal	No	Yes
wanarp	Remote Access IP ARP Driver		
	c:\windows\system32\drivers\wanarp.sys	Yes	Manual
	Kernel Driver	Yes	Running
	Normal	No	Yes
wdica	WDICA	Not Available	Kernel Driver
	No	Manual	Stopped
	Ignore	No	OK
wlbs	Network Load Balancing		
	c:\windows\system32\drivers\wlbs.sys	No	Manual
	Kernel Driver	No	Stopped
	Normal	No	No
[Signed Drivers]			
Device Name	Signed	Device Class	
	Driver Version	Driver Date	
	Manufacturer	INF Name	Driver Name
	Device ID		
Not Available	Not Available	Not Available	Not Available
Available	Not Available	Not Available	Not Available
	HTREE\ROOT\0		
ACPI Multiprocessor PC	No	COMPUTER	
	5.2.3790.0	10/1/2002	(Standard
computers)	hal.inf	Not Available	
	ROOT\ACPI_HAL\0000		
Microsoft	ACPI-Compliant System	No	
	SYSTEM	5.2.3790.0	10/1/2002
	Microsoft acpi.inf	Not Available	
	ACPI_HAL\PNP0C08\0		
Processor	PROCESSOR	5.2.3790.0	
	10/1/2002 (Standard processor types)		
	cpu.inf	Not Available	
	ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_4\0		
Processor	No	PROCESSOR	5.2.3790.0
	10/1/2002 (Standard processor types)		

```

cpu.inf Not Available
ACPI\GENUINEINTEL_-X86_FAMILY_15_MODEL_4_1
PCI bus No SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices)
machine.inf Not Available
ACPI\PNP0A03\2&DABA3FF&0
Intel(R) E7520 Memory Controller Hub - 3590 No
SYSTEM 6.3.0.1005 17/11/2004
Intel e7520.inf Not Available
PCI\VEN_8086&DEV_3590&SUBSYS_00000000&REV_0C\
3&61AAA01&0&00
Intel(R) E7525/E7520 Error Reporting Registers - 3591
No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_3591&SUBSYS_00000000&REV_0C\
3&61AAA01&0&001
Intel(R) E7525/E7520 PCI Express Root Port A0 -
3595 No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_3595&SUBSYS_00000000&REV_0C\
3&61AAA01&0&010
Intel(R) E7525/E7520 PCI Express Root Port A1 -
3596 No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_3596&SUBSYS_00000000&REV_0C\
3&61AAA01&0&018
Intel(R) 6700PXH PCI Express-to-PCI Bridge A - 0329
No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_0329&SUBSYS_00000000&REV_09\
4&d31A1BE&0&0018
DEC 21154 PCI to PCI bridge No SYSTEM
5.2.3790.0 10/1/2002 DEC
machine.inf Not Available
PCI\VEN_1011&DEV_0026&SUBSYS_00000000&REV_05\
5&d4388E4&0&080018
Mylex extremeRAID 2000 Disk Array Controller
(Accelerated) No SC5IADAPTER
7.0.14.0 9/9/2002 Mylex oem5.inf Not
Available PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\
6&360400C6&0&40080218
SCSI Enclosure Services Processor No
SYSTEM 1.0.0.0 11/8/2001 Eurologic
oem1.inf Not Available
SCSI\PROCESSOR&VEN_EUROLOGC&PROD_ULTRABLOC&RE
V_0017\7&280DC1BE&0&0F0
SCSI Enclosure Services Processor No
SYSTEM 1.0.0.0 11/8/2001 Eurologic
oem1.inf Not Available
SCSI\PROCESSOR&VEN_EUROLOGC&PROD_ULTRABLOC&RE
V_0017\7&280DC1BE&0&03F0
Mylex RAID Disk Device No DISKDRIVE
5.2.3790.0 10/1/2002 Mylex
disk.inf Not Available
SCSI\DISK&VEN_MXTOR&PROD_ATLAS15K_36SCA&REV_
_0702\7&31E1E9FE&0&400
Mylex GAM Device No SYSTEM 5.2.3790.0
10/1/2002 Mylex scsudev.inf Not
Available SCSI\PROCESSOR&VEN_MYLEX&PROD_GAM_DEVICE&REV_
\7&31E1E9FE&0&660
Adaptec AIC-7902B - Ultra320 SCSI No
SC5IADAPTER 3.0.0.0 28/8/2003
Adaptec oem0.inf Not Available
PCI\VEN_9005&DEV_801D&SUBSYS_548015D9&REV_10\
5&d4388E4&0&100018
Mylex extremeRAID 2000 Disk Array Controller
(Accelerated) No SC5IADAPTER
7.0.14.0 9/9/2002 Mylex oem5.inf Not
Available PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\
6&3057A610&0&40180018
SCSI Enclosure Services Processor No
SYSTEM 1.0.0.0 11/8/2001 Eurologic
oem1.inf Not Available
SCSI\PROCESSOR&VEN_EUROLOGC&PROD_ULTRABLOC&RE
V_0017\7&1BE22936&0&0F0
SCSI Enclosure Services Processor No
SYSTEM 1.0.0.0 11/8/2001 Eurologic
oem1.inf Not Available
SCSI\PROCESSOR&VEN_EUROLOGC&PROD_ULTRABLOC&RE
V_0017\7&1BE22936&0&03F0
Mylex RAID Disk Device No DISKDRIVE
5.2.3790.0 10/1/2002 Mylex
disk.inf Not Available
SCSI\DISK&VEN_MYLEX&PROD_EXTREMERAIID_2000&REV
_0702\7&1BE22936&0&400
Mylex GAM Device No SYSTEM 5.2.3790.0
10/1/2002 Mylex scsudev.inf Not
Available SCSI\PROCESSOR&VEN_MYLEX&PROD_GAM_DEVICE&REV_
\7&1BE22936&0&660
Intel(R) 6700/6702PXH I/OxAPIC Interrupt Controller A -
0326 No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_0326&SUBSYS_548015D9&REV_09\
4&d31A1BE&0&0118
Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A
No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_032A&SUBSYS_00000000&REV_09\
4&d31A1BE&0&0218
DEC 21154 PCI to PCI bridge No SYSTEM
5.2.3790.0 10/1/2002 DEC
machine.inf Not Available
PCI\VEN_1011&DEV_0026&SUBSYS_00000000&REV_05\
5&2199E002&0&080218
Mylex extremeRAID 2000 Disk Array Controller
(Accelerated) No SC5IADAPTER
7.0.14.0 9/9/2002 Mylex oem5.inf Not
Available

```

```

Disk drive No DISKDRIVE 5.2.3790.0
10/1/2002 (Standard disk drives)
disk.inf Not Available
SCSI\DISK&VEN_MXTOR&PROD_ATLAS15K_36SCA&REV_
DT60\6&35528202&0&000
Disk drive No DISKDRIVE 5.2.3790.0
10/1/2002 (Standard disk drives)
disk.inf Not Available
SCSI\DISK&VEN_MXTOR&PROD_ATLAS15K_36SCA&REV_
DT60\6&35528202&0&010
SCSI Backplane Device No SYSTEM Not
Available Not Available SUPER MICRO COMPUTER,
INC. oem3.inf Not Available
SCSI\PROCESSOR&VEN_SUPER&PROD_GEM359_REV001&R
EV_1.09\6&35528202&0&080
Adaptec AIC-7902B - Ultra320 SCSI No
SC5IADAPTER 3.0.0.0 28/8/2003
Adaptec oem0.inf Not Available
PCI\VEN_9005&DEV_801D&SUBSYS_548015D9&REV_10\
5&d4388E4&0&110018
DEC 21154 PCI to PCI bridge No SYSTEM
5.2.3790.0 10/1/2002 DEC
machine.inf Not Available
PCI\VEN_1011&DEV_0026&SUBSYS_00000000&REV_05\
5&d4388E4&0&180018
Mylex extremeRAID 2000 Disk Array Controller
(Accelerated) No SC5IADAPTER
7.0.14.0 9/9/2002 Mylex oem5.inf Not
Available PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\
6&3057A610&0&40180018
SCSI Enclosure Services Processor No
SYSTEM 1.0.0.0 11/8/2001 Eurologic
oem1.inf Not Available
SCSI\PROCESSOR&VEN_EUROLOGC&PROD_ULTRABLOC&RE
V_0017\7&1BE22936&0&0F0
SCSI Enclosure Services Processor No
SYSTEM 1.0.0.0 11/8/2001 Eurologic
oem1.inf Not Available
SCSI\PROCESSOR&VEN_EUROLOGC&PROD_ULTRABLOC&RE
V_0017\7&1BE22936&0&03F0
Mylex RAID Disk Device No DISKDRIVE
5.2.3790.0 10/1/2002 Mylex
disk.inf Not Available
SCSI\DISK&VEN_MYLEX&PROD_EXTREMERAIID_2000&REV
_0702\7&1BE22936&0&400
Mylex GAM Device No SYSTEM 5.2.3790.0
10/1/2002 Mylex scsudev.inf Not
Available SCSI\PROCESSOR&VEN_MYLEX&PROD_GAM_DEVICE&REV_
\7&1BE22936&0&660
Intel(R) 6700/6702PXH I/OxAPIC Interrupt Controller A -
0326 No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_0326&SUBSYS_548015D9&REV_09\
4&d31A1BE&0&0118
Intel(R) 6700PXH PCI Express-to-PCI Bridge B - 032A
No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_032A&SUBSYS_00000000&REV_09\
4&d31A1BE&0&0218
DEC 21154 PCI to PCI bridge No SYSTEM
5.2.3790.0 10/1/2002 DEC
machine.inf Not Available
PCI\VEN_1011&DEV_0026&SUBSYS_00000000&REV_05\
5&2199E002&0&080218
Mylex extremeRAID 2000 Disk Array Controller
(Accelerated) No SC5IADAPTER
7.0.14.0 9/9/2002 Mylex oem5.inf Not
Available

```

```

PCI\VEN_1069&DEV_BA56&SUBSYS_00401069&REV_00\
6&360400C6&0&40080218
SCSI Enclosure Services Processor No
SYSTEM 1.0.0.0 11/8/2001 Eurologic
oem1.inf Not Available
SCSI\PROCESSOR&VEN_EUROLOGC&PROD_ULTRABLOC&RE
V_0017\7&31E1E9FE&0&0F0
Mylex RAID Disk Device No DISKDRIVE
5.2.3790.0 10/1/2002 Mylex
disk.inf Not Available
SCSI\DISK&VEN_MYLEX&PROD_EXTREMERAIID_2000&REV
_0702\7&31E1E9FE&0&400
Mylex GAM Device No SYSTEM 5.2.3790.0
10/1/2002 Mylex scsudev.inf Not
Available SCSI\PROCESSOR&VEN_MYLEX&PROD_GAM_DEVICE&REV_
\7&31E1E9FE&0&660
Intel(R) PRO/1000 MT Dual Port Server Adapter No
NET 8.4.21.0 22/11/2004 Intel
oem2.inf Not Available
PCI\VEN_8086&DEV_1079&SUBSYS_117A8086&REV_03\
5&2199E002&0&100218
Intel(R) PRO/1000 MT Dual Port Server Adapter No
NET 8.4.21.0 22/11/2004 Intel
oem2.inf Not Available
PCI\VEN_8086&DEV_1079&SUBSYS_117A8086&REV_03\
5&2199E002&0&110218
Intel(R) 6700PXH I/OxAPIC Interrupt Controller B - 0327
No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_0327&SUBSYS_548015D9&REV_09\
4&d31A1BE&0&0318
Intel(R) E7525/E7520 PCI Express Root Port B0 - 3597
No SYSTEM 6.3.0.1005
17/11/2004 Intel e7520.inf Not
Available PCI\VEN_8086&DEV_3597&SUBSYS_00000000&REV_0C\
3&61AAA01&0&020
Intel(R) E7520 PCI Express Root Port C0 - 3599 No
SYSTEM 6.3.0.1005 17/11/2004
Intel e7520.inf Not Available
PCI\VEN_8086&DEV_3599&SUBSYS_00000000&REV_0C\
3&61AAA01&0&030
Intel(R) 82801EB USB Universal Host Controller - 24D2
No USB 6.3.0.1005
17/11/2004 Intel ich5usb.inf
Not Available
PCI\VEN_8086&DEV_24D2&SUBSYS_548015D9&REV_02\
3&61AAA01&0&0E8
USB Root Hub No USB 5.2.3790.0
10/1/2002 (Standard USB Host Controller)
usbport.inf Not Available
USB\ROOT_HUB\4&10B6329E&0
Intel(R) 82801EB USB Universal Host Controller - 24D4
No USB 6.3.0.1005
17/11/2004 Intel ich5usb.inf
Not Available
PCI\VEN_8086&DEV_24D4&SUBSYS_548015D9&REV_02\
3&61AAA01&0&0E9
USB Root Hub No USB 5.2.3790.0
10/1/2002 (Standard USB Host Controller)
usbport.inf Not Available
USB\ROOT_HUB\4&8949A15D&0
Intel(R) 82801EB USB Universal Host Controller - 24D7
No USB 6.3.0.1005
17/11/2004 Intel ich5usb.inf
Not Available
PCI\VEN_8086&DEV_24D7&SUBSYS_548015D9&REV_02\
3&61AAA01&0&0EA
USB Root Hub No USB 5.2.3790.0
10/1/2002 (Standard USB Host Controller)

```

```

usbport.inf Not Available
USB\ROOT_HUB\4&33788D83&0
Intel(R) 82801EB USB Universal Host Controller - 24DE No USB 6.3.0.1005
17/11/2004 Intel ich5usb.inf
Not Available
PCI\VEN_8086&DEV_24DE&SUBSYS_548015D9&REV_02\
3&61AAA01&0&EB
USB Root Hub No USB 5.2.3790.0
10/1/2002 (Standard USB Host Controller)
usbport.inf Not Available
USB\ROOT_HUB\4&353EC4EB&0
Intel(R) 82801EB USB Enhanced Host Controller - 24DD No USB 6.3.0.1005
17/11/2004 Intel ich5usb.inf
Not Available
PCI\VEN_8086&DEV_24DD&SUBSYS_548015D9&REV_02\
3&61AAA01&0&EF
USB Root Hub No USB 5.2.3790.0
10/1/2002 (Standard USB Host Controller)
usbport.inf Not Available
USB\ROOT_HUB2\4&2B858A8E&0
USB Mass Storage Device No USB 5.2.3790.0
10/1/2002 Compatible USB
storage device usbstor.inf Not Available
USB\VID_0EAO&PID_2168\382F41ACD2950022
Disk drive No DISKDRIVE 5.2.3790.0
10/1/2002 (Standard disk drives)
disk.inf Not Available
USBSTOR\DISK&VEN_FLASH&PROD_DRIVE_OT_USB20&RE
V_2_00\382F41ACD2950022&0
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\REMOVABLEMEDIA\7&EAF53AF&0&RM
Intel(R) 82801 PCI Bridge - 244E No
SYSTEM 7.0.0.1011 1/10/2005 Intel
dmi_pci.inf Not Available
PCI\VEN_8086&DEV_244E&SUBSYS_00000000&REV_C2\
3&61AAA01&0&F0
RAGE XL PCI Family (Microsoft Corporation) No
DISPLAY 5.10.2600.6014 8/8/2001 ATI
Technologies Inc. atiixpad.inf Not Available
PCI\VEN_1002&DEV_4752&SUBSYS_80081002&REV_27\
4&3A321F38&0&08F0
Plug and Play Monitor No MONITOR
5.1.2001.0 (Standard
monitor types) monitor.inf Not Available
DISPLAY\GSM4386\5&22A9F974&0&8000001&0A&01
Intel(R) 82801EB LPC Interface Controller - 24D0 No
SYSTEM 7.0.0.1011 1/10/2005 Intel
ich5core.inf Not Available
PCI\VEN_8086&DEV_24D0&SUBSYS_00000000&REV_02\
3&61AAA01&0&F8
ISAPNP Read Data Port No SYSTEM
5.2.3790.0 10/1/2002 (Standard
system devices) machine.inf Not Available
ISAPNP\READDATAPORT\0
Motherboard resources No SYSTEM
5.2.3790.0 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0C02\1F
Direct memory access controller No
SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf
Not Available ACPI\PNP0200\4&369939D9&0
Numeric data processor No SYSTEM
5.2.3790.0 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0C04\4&369939D9&0

```

```

Programmable interrupt controller No
SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf
Not Available ACPI\PNP0000\4&369939D9&0
System CMOS/real time clock No SYSTEM
5.2.3790.0 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\PNP0B00\4&369939D9&0
System speaker No SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices)
machine.inf Not Available
ACPI\PNP0800\4&369939D9&0
System timer No SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices)
machine.inf Not Available
ACPI\PNP0100\4&369939D9&0
Intel(r) 82802 Firmware Hub Device No
SYSTEM 5.2.3790.0 10/1/2002 Intel
machine.inf Not Available
ACPI\INT0800\4&369939D9&0
Generic Bus No SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices)
machine.inf Not Available
ACPI\PNP0A05\4&369939D9&0
ECP Printer Port No PORTS 5.2.3790.0
10/1/2002 (Standard port types)
msports.inf Not Available
ACPI\PNP0401\5&18D3520B&0
Printer Port Logical Interface No
SYSTEM 5.2.3790.0 10/1/2002
(Standard system devices) machine.inf
Not Available
LPTENUM\MICROSOFTRAWPORT\6&728A26C&0&LPT1
Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
No KEYBOARD 5.2.3790.0
10/1/2002 (Standard keyboards)
keyboard.inf Not Available
ACPI\PNP0303\5&18D3520B&0
PS/2 Compatible Mouse No MOUSE
5.2.3790.0 10/1/2002 Microsoft
msmouse.inf Not Available
ACPI\PNP0F13\5&18D3520B&0
Standard floppy disk controller No FDC
5.2.3790.0 10/1/2002 (Standard
floppy disk controllers) fdc.inf Not Available
ACPI\PNP0700\5&18D3520B&0
Floppy disk drive No FLOPPYDISK
5.2.3790.0 10/1/2002 (Standard
floppy disk drives) floydisk.inf Not Available
FDC\GENERIC_FLOPPY_DRIVE\6&297D0BE5&0&0
Intel(R) 82801EB Ultra ATA Storage Controllers No
HDC 6.3.0.1005 17/11/2004
Intel ich5ide.inf Not Available
PCI\VEN_8086&DEV_24DB&SUBSYS_548015D9&REV_02\
3&61AAA01&0&F9
Primary IDE Channel No HDC 5.2.3790.0
10/1/2002 (Standard IDE ATA/ATAPI
controllers) mshdc.inf Not Available
PCIIDE\IDECHANNEL\4&7145BEC&0&0
CD-ROM Drive No CDROM 5.2.3790.0
10/1/2002 (Standard CD-ROM drives)
cdrom.inf Not Available
IDE\CDROMSONLY_CD-
ROM_CDU5212_5ys1_5&6643C9C&0&0.
0.0
Secondary IDE Channel No HDC
5.2.3790.0 10/1/2002 (Standard IDE
ATA/ATAPI controllers) mshdc.inf Not Available
PCIIDE\IDECHANNEL\4&7145BEC&0&01

```

```

Intel(R) 82801EB SMBus Controller - 24D3 No
SYSTEM 7.0.0.1011 1/10/2005 Intel
ich5core.inf Not Available
PCI\VEN_8086&DEV_24D3&SUBSYS_548015D9&REV_02\
3&61AAA01&0&FB
ACPI Power Button No SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices)
machine.inf Not Available
ACPI\PNP0C0C\3&61AAA01&0
ACPI Fixed Feature Button No SYSTEM
5.2.3790.0 10/1/2002 (Standard
system devices) machine.inf Not Available
ACPI\FIXEDBUTTON\2&DABA3FF&0
Logical Disk Manager No SYSTEM
5.2.3790.0 10/1/2002 (Standard
system devices) machine.inf Not Available
ROOT\DMIO\0000
Volume Manager No SYSTEM 5.2.3790.0
10/1/2002 (Standard system devices)
machine.inf Not Available
ROOT\FTDISK\0000
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREEFC13A28
OFFSET7E00LENGTHEA608C400
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREEFC13A28
OFFSETEA6094200LENGTHIADB799000
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREEFC13A28
OFFSET298182D200LENGTHH9EB14F6000
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREB545B545
OFFSET7E00LENGTH88E02E800
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATURE93739373
OFFSET7E00LENGTH88D856600
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREEFC13A37
OFFSETEA6094200LENGTHIADB799000
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREEFC13A37
OFFSETEA6094200LENGTHIADB799000
Generic volume No VOLUME 5.2.3790.0
10/1/2002 Microsoft volume.inf Not
Available STORAGE\VOLUME\1&30A96598&0&SIGNATUREE07C3360
OFFSET7E00LENGTH22A52E200
AFD Networking Support Environment Not Available
LEGACYDRIVER Not Available
Not Available Not Available Not
Available Not Available Not Available Not
Available ROOT\LEGACY_AFD\0000
Beep Not Available LEGACYDRIVER Not
Available Not Available Not Available

```

Available	Not Available	ROOT\LEGACY_BEEP\0000		
CRC Disk Filter Driver	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_CRCDISK\0000		Not
dmboot	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_DMBOOT\0000		Not
dmload	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_DMLOAD\0000		Not
Fips	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_FIPS\0000		Not
Generic Packet Classifier	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_GPC\0000		Not
IntelIde	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_INTELIDE\0000		Not
IPSEC driver	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_IPSEC\0000		Not
ksecdd	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_KSECCD\0000		Not
mac2w2k	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_MAC2W2K\0000		Not
mmdd	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_MMDD\0000		Not
mountmgr	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_MOUNTMGR\0000		Not
NDIS System Driver	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_NDIS\0000		Not
Remote Access NDIS TAPI Driver	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_NDIS\0000		Not
NDIS Usermode I/O Protocol	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_NDIS\0000		Not
NDProxy	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_NDPROXY\0000		Not
NetBios over Tcpip	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_NETBT\0000		Not
Null	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_NULL\0000		Not
Partition Manager	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_PARTITION\0000		Not

Available	Not Available	ROOT\LEGACY_PARTMGR\0000		Not Available
Parvdm	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_PARVDM\0000		Not
Remote Access Auto Connection Driver	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_RASACD\0000		Not
RDPCCD	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_RDPCCD\0000		Not
TCP/IP Protocol Driver	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_TCPIP\0000		Not
VGA Display Controller	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_VGASAVE\0000		Not
volsnap	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_VOLSNAP\0000		Not
Remote Access IP ARP Driver	Not Available	LEGACYDRIVER	Not Available	Not
Available	Not Available	ROOT\LEGACY_WANARP\0000		Not
Audio Codecs	No MEDIA	5.2.3790.0	10/1/2002 (Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMCMAC		
Legacy Audio Drivers	No MEDIA	5.2.3790.0	10/1/2002 (Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMDRV		
Media Control Devices	No MEDIA	5.2.3790.0	10/1/2002 (Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMMCI		
Legacy Video Capture Devices	No MEDIA	5.2.3790.0	10/1/2002 (Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMVCD		
Video Codecs	No MEDIA	5.2.3790.0	10/1/2002 (Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMVID		
WAN Miniport (L2TP)	No NET	5.2.3790.0	10/1/2002 Microsoft netrasa.inf	Not
Available	Not Available	ROOT\MS_L2TPMINIPORT\0000		Not
WAN Miniport (IP)	No NET	5.2.3790.0	10/1/2002 Microsoft netrasa.inf	Not
Available	Not Available	ROOT\MS_NDISWANIP\0000		Not
WAN Miniport (PPPOE)	No NET	5.2.3790.0	10/1/2002 Microsoft netrasa.inf	Not
Available	Not Available	ROOT\MS_PPPOEMINIPORT\0000		Not
WAN Miniport (PPTP)	No NET	5.2.3790.0	10/1/2002 Microsoft netrasa.inf	Not
Available	Not Available	ROOT\MS_PPTPMINIPORT\0000		Not
Direct Parallel	No NET	5.2.3790.0	10/1/2002 Microsoft netrasa.inf	Not
Available	Not Available	ROOT\MS_PTMINIPORT\0000		Not
Terminal Server Device Redirector	No SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf
Not Available	Not Available	ROOT\RDPDR\0000		
Terminal Server Keyboard Driver	No SYSTEM	5.2.3790.0	10/1/2002	

(Standard system devices)	machine.inf			
Not Available	ROOT\RDP_KBD\0000			
Terminal Server Mouse Driver	No SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf
Not Available	ROOT\RDP_MOUSE\0000			
Plug and Play Software Device Enumerator	No SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf
Not Available	ROOT\SYSTEM\0000			
Microcode Update Device	No SYSTEM	5.2.3790.0	10/1/2002 (Standard system devices)	machine.inf
Not Available	ROOT\SYSTEM\0001			
[Environment variables]				
Variable	Value	User Name		
Comspec	%SystemRoot%\system32\cmd.exe	<SYSTEM>		
Path	%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\system32\wbem;C:\Program Files\Microsoft SQL Server\80\Tools\BINN	<SYSTEM>		
t%	%SystemRoot%	<SYSTEM>		
windir	OS	windows_NT		
OS	PROCESSOR_ARCHITECTURE	x86	<SYSTEM>	
PROCESSOR_ARCHITECTURE	PROCESSOR_LEVEL	15	<SYSTEM>	
PROCESSOR_LEVEL	PROCESSOR_IDENTIFIER	x86 Family 15 Model 4 Stepping 3, GenuineIntel	<SYSTEM>	
PROCESSOR_IDENTIFIER	PROCESSOR_REVISION	0403	<SYSTEM>	
PROCESSOR_REVISION	NUMBER_OF_PROCESSORS	2	<SYSTEM>	
NUMBER_OF_PROCESSORS	ClusterLog	C:\WINDOWS\cluster\cluster.log		
ClusterLog	<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	NT		
AUTHORITY\SYSTEM	TMP	%USERPROFILE%\Local Settings\Temp	NT	
AUTHORITY\SYSTEM	TEMP	%USERPROFILE%\Local Settings\Temp		
TEMP	SQL3040\Administrator			
SQL3040\Administrator	TEMP	%USERPROFILE%\Local Settings\Temp		
%USERPROFILE%\Local Settings\Temp	[Print Jobs]			
Document	Size	Owner	Notify	Status
Submitted	Start Time	Until Time	Pages Printed	Job
ID	Elapsed Time	Priority	Parameters	Driver
Processor	Host Print Queue	Data Type	Name	Print
[Network Connections]				
Local Name	Status	Remote Name	User Name	Type
[Running Tasks]				
Name	Path	Process ID	Priority	Min
Working Set	Version	Max Working Set	File Date	Start Time
system	idle process	Not Available	0	0
Available	Not Available	Not Available	Not Available	Not
Available	Not Available	Not Available	Not Available	Not
Available	Not Available	Not Available	Not Available	Not
system	Not Available	4	8	0
1413120	Not Available	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	Not Available	Not Available

```

smss.exe Not Available 488 11
204800 1413120 17/6/2005 16:46 Not
Available Not Available
csrss.exe Not Available 544 13 Not
Available Not Available 17/6/2005 16:46 Not
Available Not Available Not Available
winlogon.exe c:\windows\system32\winlogon.exe
572 13 204800 1413120
17/6/2005 16:46 5.2.3790.0
(srv03_rtm.030324-2048) 536,50 KB (549.376 bytes)
25/3/2003 09:00
services.exe c:\windows\system32\services.exe
616 9 204800 1413120
17/6/2005 16:46 5.2.3790.0
(srv03_rtm.030324-2048) 102,00 KB (104.448 bytes)
25/3/2003 09:00
lsass.exe c:\windows\system32\lsass.exe 628 9
204800 1413120 17/6/2005 16:46
5.2.3790.0 (srv03_rtm.030324-2048) 13,00
KB (13.312 bytes) 25/3/2003 09:00
svchost.exe c:\windows\system32\svchost.exe
824 8 204800 1413120
17/6/2005 16:46 5.2.3790.0
(srv03_rtm.030324-2048) 13,00 KB (13.312 bytes)
25/3/2003 09:00
svchost.exe c:\windows\system32\svchost.exe
872 8 204800 1413120
17/6/2005 16:46 5.2.3790.0
(srv03_rtm.030324-2048) 13,00 KB (13.312 bytes)
25/3/2003 09:00
svchost.exe c:\windows\system32\svchost.exe
992 8 204800 1413120
17/6/2005 16:46 5.2.3790.0
(srv03_rtm.030324-2048) 13,00 KB (13.312 bytes)
25/3/2003 09:00
wmiprvse.exe Not Available 1160 8
Not Available Not Available
17/6/2005 16:48 Not Available Not
Available Not Available
explorer.exe c:\windows\explorer.exe 1368
8 204800 1413120 17/6/2005 16:50
6.00.3790.0 (srv03_rtm.030324-2048)
1.008,50 KB (1.032.704 bytes) 25/3/2003 09:00

ctfmon.exe c:\windows\system32\ctfmon.exe
1448 8 204800 1413120
17/6/2005 16:50 5.2.3790.0
(srv03_rtm.030324-2048) 13,50 KB (13.824 bytes)
25/3/2003 09:00
sqlmangr.exe c:\program files\microsoft sql
server\80\tools\bin\sqlmangr.exe 1464 8
204800 1413120 17/6/2005 16:50
2000.080.0760.00 72,57 KB (74.308 bytes)
1/6/2005 18:09
wpabaln.exe c:\windows\system32\wpabaln.exe
1608 8 204800 1413120
17/6/2005 16:52 5.2.3790.0
(srv03_rtm.030324-2048) 31,00 KB (31.744 bytes)
25/3/2003 09:00
cmd.exe c:\windows\system32\cmd.exe 1920 8
204800 1413120 20/6/2005 08:30
5.2.3790.0 (srv03_rtm.030324-2048)
374,00 KB (382.976 bytes) 25/3/2003 09:00

sqlservr.exe c:\program files\microsoft sql
server\mssql\bin\sqlservr.exe 1940 13
204800 1413120 20/6/2005 08:30
2000.080.0760.00 7,17 MB (7.520.337 bytes)
1/6/2005 18:08
isqlw.exe c:\program files\microsoft sql
server\80\tools\bin\isqlw.exe 1296 8
204800 1413120 20/6/2005 08:35

```

```

2000.080.0760.00 344,56 KB (352.828 bytes)
1/6/2005 18:08
cmd.exe c:\windows\system32\cmd.exe 380 8
204800 1413120 20/6/2005 11:23
5.2.3790.0 (srv03_rtm.030324-2048)
374,00 KB (382.976 bytes) 25/3/2003 09:00

helpctr.exe
exe c:\windows\pchealth\helpctr\binaries\helpctr.
1576 8 204800 1413120
20/6/2005 11:50 5.2.3790.0
(srv03_rtm.030324-2048) 764,00 KB (782.336 bytes)
27/5/2005 11:35
wmiprvse.exe Not Available 368 8
Not Available Not Available
20/6/2005 11:50 Not Available Not
Available Not Available
helpsvc.exe c:\windows\pchealth\helpctr\binaries\helpsvc.
exe 1712 8 204800 1413120
20/6/2005 11:50 5.2.3790.0
(srv03_rtm.030324-2048) 720,00 KB (737.280 bytes)
27/5/2005 11:35

[Loaded Modules]
Name Version Size File Date Manufacturer
Path
winlogon 5.2.3790.0 (srv03_rtm.030324-2048)
536,50 KB (549.376 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\winlogon.exe
ntdll 5.2.3790.0 (srv03_rtm.030324-2048)
722,50 KB (739.840 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\ntdll.dll
kernel32 5.2.3790.0 (srv03_rtm.030324-2048)
965,00 KB (988.160 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\kernel32.dll
msvcrt 7.0.3790.0 (srv03_rtm.030324-2048)
319,50 KB (327.168 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\msvcrt.dll
advapi32 5.2.3790.0 (srv03_rtm.030324-2048)
559,50 KB (572.928 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\advapi32.dll
rpcrt4 5.2.3790.0 (srv03_rtm.030324-2048)
643,50 KB (658.944 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\rpcrt4.dll
user32 5.2.3790.0 (srv03_rtm.030324-2048)
562,00 KB (575.488 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\user32.dll
gdi32 5.2.3790.0 (srv03_rtm.030324-2048)
263,00 KB (269.312 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\gdi32.dll
userenv 5.2.3790.0 (srv03_rtm.030324-2048)
732,50 KB (750.080 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\userenv.dll
nddeapi 5.2.3790.0 (srv03_rtm.030324-2048)
16,00
KB (16.384 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\nddeapi.dll

crypt32 5.131.3790.0 (srv03_rtm.030324-2048)
598,00 KB (612.352 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\crypt32.dll

```

```

msasn1 5.2.3790.0 (srv03_rtm.030324-2048) 58,00
KB (59.392 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\msasn1.dll

secur32 5.2.3790.0 (srv03_rtm.030324-2048) 63,00
KB (64.512 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\secur32.dll

winsta 5.2.3790.0 (srv03_rtm.030324-2048) 51,00
KB (52.224 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\winsta.dll

netapi32 5.2.3790.0 (srv03_rtm.030324-2048)
317,00 KB (324.608 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\netapi32.dll
profmap 5.2.3790.0 (srv03_rtm.030324-2048) 22,00
KB (22.528 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\profmap.dll

regapi 5.2.3790.0 (srv03_rtm.030324-2048) 48,50
KB (49.664 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\regapi.dll

ws2_32 5.2.3790.0 (srv03_rtm.030324-2048) 87,50
KB (89.600 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\ws2_32.dll

ws2help 5.2.3790.0 (srv03_rtm.030324-2048) 19,50
KB (19.968 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\ws2help.dll

psapi 5.2.3790.0 (srv03_rtm.030324-2048) 21,50
KB (22.016 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\psapi.dll
version 5.2.3790.0 (srv03_rtm.030324-2048) 17,00
KB (17.408 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\version.dll

setupapi 5.2.3790.0 (srv03_rtm.030324-2048)
1.014,50 KB (1.038.848 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\setupapi.dll
msgina 5.2.3790.0 (srv03_rtm.030324-2048) 1,14
MB (1.191.936 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\msgina.dll
shsvcs 6.00.3790.0 (srv03_rtm.030324-2048)
121,50 KB (124.416 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\shsvcs.dll
shlwapi 6.00.3790.0 (srv03_rtm.030324-2048)
281,00 KB (287.744 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\shlwapi.dll
sfc 5.2.3790.0 (srv03_rtm.030324-2048) 4,50
KB (4.608 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\sfc.dll
sfc_os 5.2.3790.0 (srv03_rtm.030324-2048)
133,00 KB (136.192 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\sfc_os.dll
wintrust 5.131.3790.0 (srv03_rtm.030324-2048)
161,50 KB (165.376 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\wintrust.dll
ole32 5.2.3790.0 (srv03_rtm.030324-2048) 1,13
MB (1.187.328 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\ole32.dll
imagehlp 5.2.3790.0 (srv03_rtm.030324-2048)
142,50 KB (145.920 bytes) 25/3/2003 09:00

```



```

Microsoft Corporation
c:\windows\system32\imagehp.dll
comctl32 6.0 (srv03_rtm.030324-2048) 907.00 KB
Corporation 27/5/2005 08:20 Microsoft
c:\windows\x86_microsoft.windows.common
n-contr15_6595b6444ccf1df_6.0.100.0_x-
ww_8417450b_comctl32.dll 98.50
wincard 5.2.3790.0 (srv03_rtm.030324-2048)
Corporation 25/3/2003 09:00 Microsoft
c:\windows\system32\wincard.dll
wtsapi32 5.2.3790.0 (srv03_rtm.030324-2048) 17.50
Corporation 25/3/2003 09:00 Microsoft
c:\windows\system32\wtsapi32.dll
sxs 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\sxs.dll
winmm 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\winmm.dll
rsaenh 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\rsaenh.dll
wldap32 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\wldap32.dll
csddl1 5.2.3790.0 (srv03_rtm.030324-2048) 99.00
Corporation 25/3/2003 09:00 Microsoft
c:\windows\system32\csddl1.dll
wlnotify 5.2.3790.0 (srv03_rtm.030324-2048) 87.50
KB (89.600 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\wlnotify.dll
winspool 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\winspool.drv
mpr 5.2.3790.0 (srv03_rtm.030324-2048) 56.00
Corporation 25/3/2003 09:00 Microsoft
c:\windows\system32\mpr.dll
shel 8.166.400 bytes 25/3/2003 09:00
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\shel132.dll
comctl32 5.82 (srv03_rtm.030324-2048) 561.00 KB
Corporation 27/5/2005 08:20 Microsoft
c:\windows\x86_microsoft.windows.common
n-contr15_6595b6444ccf1df_5.82.0.0_x-
ww_8a69ba05_comctl32.dll
uxtheme 189.00 KB (200.704 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\uxtheme.dll
mprapi 5.2.3790.0 (srv03_rtm.030324-2048) 81.00
Corporation 25/3/2003 09:00 Microsoft
c:\windows\system32\mprapi.dll
activeds 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\activeds.dll
adsldpc 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\adsldpc.dll

```

```

Microsoft Corporation
c:\windows\system32\adsldpc.dll
credui 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\credui.dll
atl 3.05.2283.83.00 KB (84.992 bytes)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\atl.dll
oleaut32 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\oleaut32.dll
rtutils 5.2.3790.0 (srv03_rtm.030324-2048) 32.00
KB (32.768 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\rtutils.dll
samlib 5.2.3790.0 (srv03_rtm.030324-2048) 49.00
KB (50.176 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\samlib.dll
cscuri 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\cscuri.dll
cibcatq 2001.12.4720.0 (srv03_rtm.030324-2048)
Microsoft Corporation 27/5/2005 11:33
c:\windows\system32\cibcatq.dll
comres 2001.12.4720.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\comres.dll
ntmarta 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\ntmarta.dll
wbemprox 5.2.3790.0 (srv03_rtm.030324-2048) 17.50
KB (17.920 bytes) 27/5/2005 11:33 Microsoft
Corporation c:\windows\system32\wbemprox.dll
wbemcomn 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\wbemcomn.dll
wbemsvc 5.2.3790.0 (srv03_rtm.030324-2048) 42.50
Corporation 27/5/2005 11:33 Microsoft
c:\windows\system32\wbem\wbemsvc.dll
fastprox 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 27/5/2005 11:33
c:\windows\system32\wbem\fastprox.dll
msvcpl60 6.05.2144.0 388.00 KB (397.312 bytes)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\msvcpl60.dll
ntdsapi 5.2.3790.0 (srv03_rtm.030324-2048) 76.00
KB (77.824 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\ntdsapi.dll
dhsapi 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\dhsapi.dll
services 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\services.exe
scesrv 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\scesrv.dll

```

```

5.2.3790.0 (srv03_rtm.030324-2048) 67.00
KB (68.608 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\unpnpmgr
unpnpmgr 121.50 KB (124.416 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\unpnpmgr.dll
ncobjapi 5.2.3790.0 (srv03_rtm.030324-2048) 34.50
KB (35.328 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\ncobjapi.dll
eventlog 5.2.3790.0 (srv03_rtm.030324-2048) 60.50
KB (61.952 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\eventlog.dll
lsass 5.2.3790.0 (srv03_rtm.030324-2048) 13.00
KB (13.312 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\lsass.exe
lsasrv 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\lsasrv.dll
samsrv 452.00 KB (462.848 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\samsrv.dll
cryptdll 5.2.3790.0 (srv03_rtm.030324-2048) 34.00
KB (34.816 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\cryptdll.dll
mspapi 5.2.3790.0 (srv03_rtm.030324-2048) 46.50
KB (47.616 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\mspapi.dll
kerberos 332.50 KB (340.480 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\kerberos.dll
msv1_0 127.00 KB (130.048 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\msv1_0.dll
netlogon 409.00 KB (418.816 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\netlogon.dll
w32time 212.00 KB (216.960 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\w32time.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50
KB (84.480 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\iphlpapi.dll
schannel 5.2.3790.0 (srv03_rtm.030324-2048)
Microsoft Corporation 25/3/2003 09:00
c:\windows\system32\schannel.dll
wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00
KB (62.464 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\wdigest.dll
rasfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50
KB (20.992 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\rasfm.dll
kdcsvc 221.00 KB (226.304 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\kdcsvc.dll
ntdsas 5.2.3790.0 (srv03_rtm.030324-2048) 1.45
MB (1.520.640 bytes) 25/3/2003 09:00

```

Microsoft Corporation			
ntdsatq	5.2.3790.0 (srv03_rtm.030324-2048)	32,00 KB (32.768 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\ntdsatq.dll			
msock	5.2.3790.0 (srv03_rtm.030324-2048)	254,00 KB (260.096 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\msock.dll			
esent	5.2.3790.0 (srv03_rtm.030324-2048)	1,01 MB (1.056.256 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\esent.dll			
scecli	5.2.3790.0 (srv03_rtm.030324-2048)	179,50 KB (183.808 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\scecli.dll			
wshtcpip	5.2.3790.0 (srv03_rtm.030324-2048)	18,00 KB (18.432 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\wshtcpip.dll			
dssenh	5.2.3790.0 (srv03_rtm.030324-2048)	131,33 KB (134.480 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\dssenh.dll			
svchost	5.2.3790.0 (srv03_rtm.030324-2048)	13,00 KB (13.312 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\svchost.exe			
rpcss	5.2.3790.0 (srv03_rtm.030324-2048)	276,50 KB (283.136 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\rpcss.dll			
termsrv	5.2.3790.0 (srv03_rtm.030324-2048)	216,50 KB (221.696 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\termsrv.dll			
icaapi	5.2.3790.0 (srv03_rtm.030324-2048)	10,50 KB (10.752 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\icaapi.dll			
mstlsapi	5.2.3790.0 (srv03_rtm.030324-2048)	104,50 KB (107.008 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\mstlsapi.dll			
dmserver	5.2.3790.0 (srv03_rtm.030324-2048)	24,00 KB (24.576 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\dmserver.dll			
es	2001.12.4720.0 (srv03_rtm.030324-2048)	221,50 KB (226.816 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\es.dll			
wmisvc	5.2.3790.0 (srv03_rtm.030324-2048)	131,00 KB (134.144 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\wmisvc.dll			
vssapi	5.2.3790.0 (srv03_rtm.030324-2048)	528,00 KB (540.672 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\vssapi.dll			
sens	5.2.3790.0 (srv03_rtm.030324-2048)	35,50 KB (36.352 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\sens.dll			
comsvcs	2001.12.4720.0 (srv03_rtm.030324-2048)	1,14 MB (1.199.616 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\comsvcs.dll			
wbemcore	5.2.3790.0 (srv03_rtm.030324-2048)	457,00 KB (467.968 bytes)	27/5/2005 11:33

Microsoft Corporation			
esscli	5.2.3790.0 (srv03_rtm.030324-2048)	235,50 KB (241.152 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\esscli.dll			
wmiutils	5.2.3790.0 (srv03_rtm.030324-2048)	90,50 KB (92.672 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\wmiutils.dll			
repdrvfs	5.2.3790.0 (srv03_rtm.030324-2048)	165,00 KB (168.960 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\repdrvfs.dll			
wmiprvsd	5.2.3790.0 (srv03_rtm.030324-2048)	405,50 KB (415.232 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\wmiprvsd.dll			
wbemess	5.2.3790.0 (srv03_rtm.030324-2048)	256,50 KB (262.656 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\wbemess.dll			
ncprov	5.2.3790.0 (srv03_rtm.030324-2048)	43,00 KB (44.032 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\ncprov.dll			
netman	5.2.3790.0 (srv03_rtm.030324-2048)	209,00 KB (214.016 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\netman.dll			
rasapi32	5.2.3790.0 (srv03_rtm.030324-2048)	227,50 KB (232.960 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\rasapi32.dll			
rasman	5.2.3790.0 (srv03_rtm.030324-2048)	56,50 KB (57.856 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\rasman.dll			
tapi32	5.2.3790.0 (srv03_rtm.030324-2048)	175,00 KB (179.200 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\tapi32.dll			
wzcsvc	5.2.3790.0 (srv03_rtm.030324-2048)	272,50 KB (279.040 bytes)	25/3/2003 09:15
Microsoft Corporation			
c:\windows\system32\wzcsvc.dll			
wmi	5.2.3790.0 (srv03_rtm.030324-2048)	6,50 KB (6.656 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\wmi.dll			
dhcpcsvc	5.2.3790.0 (srv03_rtm.030324-2048)	101,50 KB (103.936 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\dhcpcsvc.dll			
wzcsapi	5.2.3790.0 (srv03_rtm.030324-2048)	24,50 KB (25.088 bytes)	25/3/2003 09:15
Microsoft Corporation			
c:\windows\system32\wzcsapi.dll			
netshell	5.2.3790.0 (srv03_rtm.030324-2048)	1,67 MB (1.747.456 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\netshell.dll			
clusapi	5.2.3790.0 (srv03_rtm.030324-2048)	56,00 KB (57.344 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\clusapi.dll			
hnetcfg	5.2.3790.0 (srv03_rtm.030324-2048)	243,50 KB (249.344 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\hnetcfg.dll			
wininet	6.00.3790.0 (srv03_rtm.030324-2048)	609,00 KB (623.616 bytes)	25/3/2003 09:00

Microsoft Corporation			
rasdlg	5.2.3790.0 (srv03_rtm.030324-2048)	642,00 KB (657.408 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\rasdlg.dll			
rasadhlp	5.2.3790.0 (srv03_rtm.030324-2048)	6,50 KB (6.656 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\rasadhlp.dll			
srsvcs	5.2.3790.0 (srv03_rtm.030324-2048)	89,00 KB (91.136 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\srsvcs.dll			
winnr	5.2.3790.0 (srv03_rtm.030324-2048)	15,00 KB (15.360 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\winnr.dll			
pchsvc	5.2.3790.0 (srv03_rtm.030324-2048)	31,50 KB (32.256 bytes)	27/5/2005 11:35
Microsoft Corporation			
c:\windows\pchealth\helipctr\binaries\pchsvc.dll			
wbemcons	5.2.3790.0 (srv03_rtm.030324-2048)	69,00 KB (70.656 bytes)	27/5/2005 11:33
Microsoft Corporation			
c:\windows\system32\wbem\wbemcons.dll			
explorer	6.00.3790.0 (srv03_rtm.030324-2048)	1.008,50 KB (1.032.704 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\explorer.exe			
browseui	6.00.3790.0 (srv03_rtm.030324-2048)	1,01 MB (1.057.280 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\browseui.dll			
shdocvw	6.00.3790.0 (srv03_rtm.030324-2048)	1,33 MB (1.393.664 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\shdocvw.dll			
apphelp	5.2.3790.0 (srv03_rtm.030324-2048)	122,00 KB (124.928 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\apphelp.dll			
themeui	6.00.3790.0 (srv03_rtm.030324-2048)	360,50 KB (369.152 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\themeui.dll			
msimg32	5.2.3790.0 (srv03_rtm.030324-2048)	4,50 KB (4.608 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\msimg32.dll			
msutb	5.2.3790.0 (srv03_rtm.030324-2048)	180,00 KB (184.320 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\msutb.dll			
msctf	5.2.3790.0 (srv03_rtm.030324-2048)	287,00 KB (293.888 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\msctf.dll			
linkinfo	5.2.3790.0 (srv03_rtm.030324-2048)	16,50 KB (16.896 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\linkinfo.dll			
ntshui	6.00.3790.0 (srv03_rtm.030324-2048)	136,00 KB (139.264 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\ntshui.dll			
urlmon	6.00.3790.0 (srv03_rtm.030324-2048)	501,50 KB (513.536 bytes)	25/3/2003 09:00
Microsoft Corporation			
c:\windows\system32\urlmon.dll			

webcheck 6.00.3790.0 (srv03_rtm.030324-2048)
261,50 KB (267.776 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\webcheck.dll

wsock32 5.2.3790.0 (srv03_rtm.030324-2048) 22,00
KB (22.528 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\wsock32.dll

stobject 5.2.3790.0 (srv03_rtm.030324-2048)
117,50 KB (120.320 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\stobject.dll

batmeter 6.00.3790.0 (srv03_rtm.030324-2048) 28,50
KB (29.184 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\batmeter.dll

powrprof 6.00.3790.0 (srv03_rtm.030324-2048) 14,50
KB (14.848 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\powrprof.dll

printui 5.2.3790.0 (srv03_rtm.030324-2048)
536,50 KB (549.376 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\printui.dll

cfgmgr32 5.2.3790.0 (srv03_rtm.030324-2048) 17,50
KB (17.920 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\cfgmgr32.dll

drprov 5.2.3790.0 (srv03_rtm.030324-2048) 12,50
KB (12.800 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\drprov.dll

ntlman 5.2.3790.0 (srv03_rtm.030324-2048) 41,00
KB (41.984 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\ntlman.dll

netui0 5.2.3790.0 (srv03_rtm.030324-2048) 75,50
KB (77.312 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\netui0.dll

netui1 5.2.3790.0 (srv03_rtm.030324-2048)
184,00 KB (188.416 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\netui1.dll

davclnt 5.2.3790.0 (srv03_rtm.030324-2048) 23,50
KB (24.064 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\davclnt.dll

shdoclc 6.00.3790.0 (srv03_rtm.030324-2048)
588,50 KB (602.624 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\shdoclc.dll

browseic 6.00.3790.0 (srv03_rtm.030324-2048) 62,00
KB (63.488 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\browseic.dll

zipfldr 6.00.3790.0 (srv03_rtm.030324-2048)
316,00 KB (323.584 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\zipfldr.dll

actxprxy 6.00.3790.0 (srv03_rtm.030324-2048) 95,00
KB (97.280 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\actxprxy.dll

mydocs 6.00.3790.0 (srv03_rtm.030324-2048) 88,00
KB (90.112 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\mydocs.dll

ctfmon 5.2.3790.0 (srv03_rtm.030324-2048) 13,50
KB (13.824 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\ctfmon.exe

sqlmangr 2000.080.0760.00 72,57 KB (74.308 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\sqlmangr.exe

server\80\sqlunir1 2000.080.0728.00 176,56 KB (180.800 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\sqlunir1.dll

comdlg32 6.00.3790.0 (srv03_rtm.030324-2048)
261,00 KB (267.264 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\comdlg32.dll

w95scm 2000.080.0760.00 48,56 KB (49.728 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\w95scm.dll

odbc32 3.525.1022.0 (srv03_rtm.030324-2048)
232,00 KB (237.568 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\odbc32.dll

sqlsvc 2000.080.0760.00 92,56 KB (94.784 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\sqlsvc.dll

odbcbc 2000.085.1022.00 (srv03_rtm.030324-2048)
24,00 KB (24.576 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\odbcbc.dll

sqlresld 2000.080.0382.00 28,56 KB (29.248 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\sqlresld.dll

odbcint 3.525.1022.0 (srv03_rtm.030324-2048) 92,00
KB (94.208 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\odbcint.dll

resutils 5.2.3790.0 (srv03_rtm.030324-2048) 59,00
KB (60.416 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\resutils.dll

mfc42u 6.05.3014.0 960,00 KB (983.040 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\mfc42u.dll

sqlsvc 2000.080.0194.00 24,00 KB (24.576 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlsvc.rll

sqlmangr 2000.080.0194.00 96,00 KB (98.304 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlmangr.rll

wpabaln 5.2.3790.0 (srv03_rtm.030324-2048) 31,00
KB (31.744 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\wpabaln.exe

cmd 5.2.3790.0 (srv03_rtm.030324-2048)
374,00 KB (382.976 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\cmd.exe

sqlservr 2000.080.0760.00 7,17 MB (7.520.337 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\sqlservr.exe

opends60 2000.080.0194.00 24,06 KB (24.639 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\opends60.dll

ums 2000.080.0760.00 52,55 KB (53.808 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\ums.dll

sqlsort 2000.080.0760.00 576,56 KB (590.396 bytes)
1/6/2005 18:08 Microsoft Corporation

c:\program files\microsoft sql
server\mssql\bin\sqlsort.dll

msvcirt 7.0.3790.0 (srv03_rtm.030324-2048) 50,00
KB (51.200 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\msvcirt.dll

sqllevn70 2000.080.0760.00 28,00 KB (28.672 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\resources\1033\sqllevn70.rll

xolehlp 2001.12.4720.0 (srv03_rtm.030324-2048) 8,50
KB (8.704 bytes) 27/5/2005 11:33 Microsoft
Corporation c:\windows\system32\xolehlp.dll

msdtcprx 2001.12.4720.0 (srv03_rtm.030324-2048)
427,50 KB (437.760 bytes) 27/5/2005 11:33
Microsoft Corporation
c:\windows\system32\msdtcprx.dll

mtxclu 2001.12.4720.0 (srv03_rtm.030324-2048) 74,50
KB (76.288 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\mtxclu.dll

ssnetlib 2000.080.0760.00 80,56 KB (82.492 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\ssnetlib.dll

ssnmpn70 2000.080.0534.00 24,56 KB (25.148 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\ssnmpn70.dll

security 5.2.3790.0 (srv03_rtm.030324-2048) 5,50
KB (5.632 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\security.dll

ssmslpcn 2000.080.0760.00 28,56 KB (29.244 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\ssmslpcn.dll

sqloledb 2000.085.1022.00 (srv03_rtm.030324-2048)
536,00 KB (548.864 bytes) 27/5/2005 11:35
Microsoft Corporation c:\program
files\common files\system\ole db\sqloledb.dll

msdart 2.80.1022.0 (srv03_rtm.030324-2048)
164,00 KB (167.936 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\msdart.dll

msdat13 2.80.1022.0 (srv03_rtm.030324-2048) 96,00
KB (98.304 bytes) 27/5/2005 11:35 Microsoft
Corporation c:\program files\common
files\system\ole db\msdat13.dll

oledb32 2.80.1022.0 (srv03_rtm.030324-2048)
500,00 KB (512.000 bytes) 27/5/2005 11:35
Microsoft Corporation c:\program
files\common files\system\ole db\oledb32.dll

oledb32r 2.80.1022.0 (srv03_rtm.030324-2048) 68,00
KB (69.632 bytes) 27/5/2005 11:35 Microsoft
Corporation c:\program files\common
files\system\ole db\oledb32r.dll

xpstar 2000.080.0760.00 280,56 KB (287.296 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\xpstar.dll

sqlresld 2000.080.0382.00 28,56 KB (29.248 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\sqlresld.dll

sqlsvc 2000.080.0760.00 92,56 KB (94.784 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\sqlsvc.dll

w95scm 2000.080.0760.00 48,56 KB (49.728 bytes)
1/6/2005 18:09 Microsoft Corporation

```

c:\program files\microsoft sql
server\mssql\bin\w95scm.dll
shfolder 6.00.3790.0 (srv03_rtm.030324-2048) 23,00
KB (23.552 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\shfolder.dll

sqlsvc 2000.080.0194.00 24,00 KB (24.576 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\resources\1033\sqlsvc.r11
xpstar 2000.080.0760.00 36,00 KB (36.864 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\mssql\bin\resources\1033\xpstar.r11
isqlw 2000.080.0760.00 344,56 KB (352.828 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\isqlw.exe
sqlgui 2000.080.0760.00 444,56 KB (455.232 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\sqlgui.dll
imm32 5.2.3790.0 (srv03_rtm.030324-2048)
105,50 KB (108.032 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\imm32.dll
sqlqry 2000.080.0760.00 392,56 KB (401.984 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\sqlqry.dll
pfutil80 2000.080.0760.00 272,56 KB (279.104 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\pfutil80.dll
semsfc 2000.080.0760.00 228,56 KB (234.048 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\semsfc.dll
pfcInt80 2000.080.0760.00 404,56 KB (414.272 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\pfcInt80.dll
sqlgui 2000.080.0194.00 56,00 KB (57.344 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlgui.r11
semsfc 2000.080.0194.00 24,00 KB (24.576 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\semsfc.r11
pfcInt80 2000.080.0194.00 28,00 KB (28.672 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\pfcInt80.r11
pfutil80 2000.080.0382.00 144,00 KB (147.456 bytes)
1/6/2005 18:09 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\pfutil80.r11
sqlqry 2000.080.0194.00 180,00 KB (184.320 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlqry.r11
isqlw 2000.080.0382.00 240,00 KB (245.760 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\program files\microsoft sql
server\80\tools\bin\resources\1033\isqlw.r11
odbccp32 3.525.1022.0 (srv03_rtm.030324-2048)
100,00 KB (102.400 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\odbccp32.dll
sqllex 2000.080.0194.00 148,06 KB (151.616 bytes)
1/6/2005 18:09 Microsoft Corporation

```

```

c:\program files\microsoft sql
server\80\tools\bin\sqllex.dll
sqlsrv32 2000.085.1022.00 (srv03_rtm.030324-2048)
404,00 KB (413.696 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\sqlsrv32.dll
sqlsrv32 2000.085.1022.00 (srv03_rtm.030324-2048)
88,00 KB (90.112 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\sqlsrv32.r11
dbnetlib 2000.085.1022 (srv03_rtm.030324-2048) 76,00
KB (77.824 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\dbnetlib.dll

dbmslpcn 2000.080.0760.00 28,56 KB (29.244 bytes)
1/6/2005 18:08 Microsoft Corporation
c:\windows\system32\dbmslpcn.dll
helpctr 5.2.3790.0 (srv03_rtm.030324-2048)
764,00 KB (782.336 bytes) 27/5/2005 11:35
Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\helpctr.
exe
hcappres 5.2.3790.0 (srv03_rtm.030324-2048) 6,50
KB (6.656 bytes) 27/5/2005 11:35 Microsoft
Corporation
c:\windows\pchealth\helpctr\binaries\hcappres
.dll
itss 5.2.3790.0 (srv03_rtm.030324-2048)
119,50 KB (122.368 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\itss.dll
msxml3 8.40.9419.0 1,28 MB (1.337.344 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\msxml3.dll
pchshell 5.2.3790.0 (srv03_rtm.030324-2048)
100,50 KB (102.912 bytes) 27/5/2005 11:35
Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\pchshell
.dll
mlang 6.00.3790.0 (srv03_rtm.030324-2048)
570,00 KB (583.680 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\mlang.dll
mshtml 6.00.3790.0 (srv03_rtm.030324-2048) 2,78
MB (2.916.352 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\mshtml.dll
msimtf 5.2.3790.0 (srv03_rtm.030324-2048)
149,00 KB (152.576 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\msimtf.dll
jscript 5.6.0.8515 436,00 KB (446.464 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\jscript.dll
msls31 3.10.349.0 147,00 KB (150.528 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\msls31.dll
mshhtml 6.00.3790.0 (srv03_rtm.030324-2048)
443,50 KB (454.144 bytes) 25/3/2003 09:00
Microsoft Corporation
c:\windows\system32\mshhtml.dll
vbscript 5.6.0.8515 404,00 KB (413.696 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\vbscript.dll
mfc42 6.05.3014.0 960,00 KB (983.040 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\mfc42.dll
msinfo 5.2.3790.0 (srv03_rtm.030324-2048)
358,50 KB (367.104 bytes) 27/5/2005 11:35
Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\msinfo.d
ll

```

```

riched32 5.2.3790.0 (srv03_rtm.030324-2048) 3,50
KB (3.584 bytes) 25/3/2003 09:00 Microsoft
Corporation c:\windows\system32\riched32.dll

riched20 5.31.23.1218 406,00 KB (415.744 bytes)
25/3/2003 09:00 Microsoft Corporation
c:\windows\system32\riched20.dll
helpsvc 5.2.3790.0 (srv03_rtm.030324-2048)
720,00 KB (737.280 bytes) 27/5/2005 11:35
Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\helpsvc.
exe

[Services]

Display Name Name State Start Mode
Service Type Path Error Control
Start Name Tag ID
Alerter Alerter Stopped Disabled Share Process
c:\windows\system32\svchost.exe -k
LocalService Normal NT AUTHORITY\LocalService
0
Application Layer Gateway Service ALG
Stopped Manual Own Process
c:\windows\system32\alg.exe Normal NT
AUTHORITY\LocalService 0
Application Management AppMgmt Stopped
Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Windows Audio AudioSrv Stopped Disabled Share
Process c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Background Intelligent Transfer Service BITS
Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Computer Browser Browser Stopped Manual Share
Process c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Indexing Service Cisvc Stopped Disabled Share
Process c:\windows\system32\cisvc.exe Normal
LocalSystem 0
ClipBook ClipSrv Stopped Disabled Own Process
c:\windows\system32\clipsrv.exe
Normal LocalSystem 0
COM+ System Application COMSysApp Stopped
Manual Own Process
c:\windows\system32\dllhost.exe
/processid:{02d4b3f1-fd88-11d1-960d-00805fc79235}
Normal LocalSystem 0
Cryptographic Services CryptSvc Stopped
Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Distributed File System Dfs Stopped
Manual Own Process
c:\windows\system32\dfsvc.exe
Normal LocalSystem 0
DHCP Client Dhcp Stopped Manual Share
Process c:\windows\system32\svchost.exe -k
networkservice Normal NT
AUTHORITY\NetworkService 0
Logical Disk Manager Administrative Service
dmadmin Stopped Manual Share Process
c:\windows\system32\lsmadmin.exe /com
Normal LocalSystem 0
Logical Disk Manager dmserver Running Auto
Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0

```

DNS Client	Dnscache	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k			
networkservice	Normal	NT		
AUTHORITY\NetworkService	0			
Error Reporting Service	ERSvc	Stopped		
Manual	Share Process			
c:\windows\system32\svchost.exe -k winerr				
Ignore	LocalSystem	0		
Event Log	Eventlog	Running	Auto	Share Process
c:\windows\system32\services.exe				
Normal	LocalSystem	0		
COM+ Event System	EventSystem	Running		
Manual	Share Process			
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
Help and Support	helpsvc	Running	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs			
Normal	LocalSystem	0		
Human Interface Device Access	HidServ	Stopped		
Disabled	Share Process			
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
HTTP SSL	HTTPFilter	Stopped	Manual	Share
Process	c:\windows\system32\lsass.exe			
Normal	LocalSystem	0		
IMAPI CD-Burning COM Service	ImapiService	Stopped	Disabled	Own Process
Stopped	Disabled	Own Process		
c:\windows\system32\imapi.exe				
Normal	LocalSystem	0		
Intersite Messaging	IsmServ	Stopped	Disabled	Own
Process	c:\windows\system32\ismerv.exe			
Normal	LocalSystem	0		
Kerberos	Key Distribution Center	kdc	Stopped	Disabled
Share Process	c:\windows\system32\lsass.exe			
Normal	LocalSystem	0		
Server	lanmanserver	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs			
Normal	LocalSystem	0		
Workstation	lanmanworkstation	Stopped		
Manual	Share Process			
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
License Logging	LicenseService	Stopped		
Disabled	Own Process			
c:\windows\system32\llssrv.exe				
Normal	NT AUTHORITY\NetworkService	0		
TCP/IP NetBIOS Helper	LmHosts	Stopped		
Manual	Share Process			
c:\windows\system32\svchost.exe -k				
localservice	Normal	NT AUTHORITY\LocalService		
0				
Messenger	Messenger	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
NetMeeting	Remote Desktop Sharing	mmmsrv	Stopped	Disabled
Own Process	c:\windows\system32\mmmsrv.exe			
Normal	LocalSystem	0		
Distributed Transaction Coordinator	MSDTC	Stopped	Manual	Own Process
c:\windows\system32\msdtc.exe				
Normal	NT			
AUTHORITY\NetworkService	0			
Windows Installer	MSIServer	Stopped	Manual	Share
Process	c:\windows\system32\msiexec.exe /v			
Normal	LocalSystem	0		
MSSQLSERVER	MSSQLSERVER	Stopped		
Manual	Own Process			
c:\program-1\microso-1\mssql\bin\sqlservr.exe				
Normal	LocalSystem	0		

MSSQLServerADHelper	MSSQLServerADHelper	Stopped		
Manual	Own Process			
c:\program				
files\microsoft sql	server\80\tools\bin\sqladhlp.exe			
Normal	LocalSystem	0		
Network DDE	NetDDE	Stopped	Disabled	Share
Process	c:\windows\system32\netdde.exe			
Normal	LocalSystem	0		
Network DDE DSDM	NetDDEdsdm	Stopped		
Disabled	Share Process			
c:\windows\system32\netdde.exe				
Normal	LocalSystem	0		
Net Logon	NetLogon	Stopped	Manual	Share Process
c:\windows\system32\lsass.exe				
Normal	LocalSystem	0		
Network Connections	Netman	Running	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs			
Normal	LocalSystem	0		
Network Location Awareness (NLA)	Nla	Running	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
File Replication	Ntfrs	Stopped	Manual	Own
Process	c:\windows\system32\ntfrs.exe			
Ignore	LocalSystem	0		
NT LM Security Support Provider	NtLmSsp	Stopped	Manual	Share Process
Stopped	Manual	Share Process		
c:\windows\system32\lsass.exe				
Normal	LocalSystem	0		
Removable Storage	NtmsSvc	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs			
Normal	LocalSystem	0		
Plug and Play	PlugPlay	Running	Auto	Share
Process	c:\windows\system32\services.exe			
Normal	LocalSystem	0		
IPSEC Services	PolicyAgent	Stopped		
Manual	Share Process			
c:\windows\system32\lsass.exe				
Normal	LocalSystem	0		
Protected Storage	ProtectedStorage	Stopped		
Manual	Share Process			
c:\windows\system32\lsass.exe				
Normal	LocalSystem	0		
Remote Access Auto Connection Manager	RasAuto	Stopped	Manual	Share Process
Stopped	Manual	Share Process		
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
Remote Access Connection Manager	RasMan	Stopped	Manual	Share Process
Stopped	Manual	Share Process		
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
Remote Desktop Help Session Manager	RDSSMgr	Stopped	Manual	Own Process
c:\windows\system32\sessmgr.exe				
Normal	LocalSystem	0		
Routing and Remote Access	RemoteAccess	Stopped	Disabled	Share Process
Stopped	Disabled	Share Process		
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
Remote Registry	RemoteRegistry	Stopped		
Manual	Share Process			
c:\windows\system32\svchost.exe -k regsvc				
Normal	NT AUTHORITY\LocalService	0		
Remote Procedure Call (RPC) Locator	RpcLocator	Stopped	Manual	Own Process
Stopped	Manual	Own Process		
c:\windows\system32\locator.exe				
Normal	NT AUTHORITY\NetworkService	0		
Remote Procedure Call (RPC)	RpcSs	Running	Auto	Share Process
Share Process	c:\windows\system32\svchost -k rpcss			
Normal	LocalSystem	0		

Resultant Set of Policy Provider	RSOPProv	Stopped	Manual	Share Process
Stopped	Manual	Share Process		
c:\windows\system32\rsopprov.exe				
Normal	LocalSystem	0		
Special Administration Console Helper	sacsvr	Stopped	Manual	Share Process
Stopped	Manual	Share Process		
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
Security Accounts Manager	SamSs	Running	Auto	Share Process
Share Process	c:\windows\system32\lsass.exe			
Normal	LocalSystem	0		
Smart Card	SCardSvr	Stopped	Manual	Share
Process	c:\windows\system32\scardsvr.exe			
Ignore	NT AUTHORITY\LocalService	0		
Task Scheduler	Schedule	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs			
Normal	LocalSystem	0		
Secondary Logon	seclgdn	Stopped	Manual	Share
Process	c:\windows\system32\svchost.exe -k netsvcs			
Ignore	LocalSystem	0		
System Event Notification	SENS	Running	Auto	Share Process
Share Process	c:\windows\system32\svchost.exe -k netsvcs			
Normal	LocalSystem	0		
Internet Connection Firewall (ICF) / Internet Connection Sharing (ICS)	SharedAccess	Stopped	Disabled	Share Process
Stopped	Disabled	Share Process		
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
Shell Hardware Detection	ShellHWDetection	Stopped	Manual	Share Process
Stopped	Manual	Share Process		
c:\windows\system32\svchost.exe -k netsvcs				
Ignore	LocalSystem	0		
Print Spooler	Spooler	Stopped	Manual	Own
Process	c:\windows\system32\spoolsv.exe			
Normal	LocalSystem	0		
SQLSERVERAGENT	SQLSERVERAGENT	Stopped		
Manual	Own Process			
c:\program-1\microso-1\mssql\bin\sqlagent.exe				
Normal	LocalSystem	0		
Windows Image Acquisition (WIA)	stisvc	Stopped	Disabled	Share Process
Stopped	Disabled	Share Process		
c:\windows\system32\svchost.exe -k imgs				
Normal	NT AUTHORITY\LocalService	0		
Microsoft Software Shadow Copy Provider	swprv	Stopped	Manual	Own Process
Stopped	Manual	Own Process		
c:\windows\system32\svchost.exe -k swprv				
Normal	LocalSystem	0		
Performance Logs and Alerts	SysmonLog	Stopped		
Manual	Own Process			
c:\windows\system32\smlogsvc.exe				
Normal	NT Authority\NetworkService	0		
Telephony	TapiSrv	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k tapisrv				
Normal	LocalSystem	0		
Terminal Services	TermService	Running		
Manual	Share Process			
c:\windows\system32\svchost.exe -k termsvc				
Normal	LocalSystem	0		
Themes	Themes	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs				
Normal	LocalSystem	0		
Telnet	TlntSvr	Stopped	Disabled	Own Process
c:\windows\system32\tlntsvr.exe				
Normal	NT AUTHORITY\LocalService	0		
Distributed Link Tracking Server	TrkSvr	Stopped	Disabled	Share Process
Stopped	Disabled	Share Process		

```

c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Distributed Link Tracking Client Trkwks
Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Terminal Services Session Directory Tssdis
Stopped Disabled Own Process
c:\windows\system32\tssdis.exe
Normal LocalSystem 0
Upload Manager uploadmgr Stopped Manual Share
Process c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Uninterruptible Power Supply UPS Stopped
Manual Own Process
c:\windows\system32\ups.exe Normal NT
AUTHORITY\LocalService 0
Virtual Disk Service vds Stopped
Manual Own Process
c:\windows\system32\vds.exe Normal
LocalSystem 0
Volume Shadow Copy VSS Stopped Manual Own
Process c:\windows\system32\vssvc.exe Normal
LocalSystem 0
Windows Time w32time Stopped Manual Share
Process c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
WebClient WebClient Stopped Disabled Share Process
c:\windows\system32\svchost.exe -k
localservice Normal NT AUTHORITY\LocalService
0
WinHTTP Web Proxy Auto-Discovery Service
WinHttpAutoProxySvc Stopped Manual Share
Process c:\windows\system32\svchost.exe -k
localservice Normal NT AUTHORITY\LocalService
0
Windows Management Instrumentation winmgmt
Running Auto Share Process
c:\windows\system32\svchost.exe -k netsvcs
Ignore LocalSystem 0
Portable Media Serial Number Service wmdmPmsN
Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Windows Management Instrumentation Driver Extensions
Wmi Stopped Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
WMI Performance Adapter WmiApSrv Stopped
Manual Own Process
c:\windows\system32\wbem\wmiapsrv.exe
Normal LocalSystem 0
Automatic Updates wuauserv Stopped Manual Share
Process c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0
Wireless Configuration WZCVCV Stopped
Manual Share Process
c:\windows\system32\svchost.exe -k netsvcs
Normal LocalSystem 0

```

[Program Groups]

```

Group Name Name User Name
Accessories Default User:Accessories
Accessories\Accessibility Default
User:Accessories\Accessibility Default User
Accessories\Entertainment Default
User:Accessories\Entertainment Default User

```

```

Startup Default User:Startup Default User
Accessories All Users:Accessories All
Users
Accessories\Accessibility All
Users:Accessories\Accessibility All Users
Accessories\Communications All
Users:Accessories\Communications All Users
Accessories\Entertainment All
Users:Accessories\Entertainment All Users
Accessories\System Tools All
Users:Accessories\System Tools All Users
Administrative Tools All Users:Administrative
Tools All Users
Microsoft SQL Server All Users:Microsoft SQL
Server All Users
Startup All Users:Startup All Users
Accessories NT AUTHORITY\SYSTEM:Accessories
NT AUTHORITY\SYSTEM
Accessories\Accessibility NT
AUTHORITY\SYSTEM:Accessories\Accessibility NT
AUTHORITY\SYSTEM
Accessories\Entertainment NT
AUTHORITY\SYSTEM:Accessories\Entertainment
NT AUTHORITY\SYSTEM
Startup NT AUTHORITY\SYSTEM:Startup NT
AUTHORITY\SYSTEM
Accessories SQL3040\Administrator:Accessories
Administrator
Accessories\Accessibility
SQL3040\Administrator:Accessories\Accessibili
ty
SQL3040\Administrator
Accessories\Entertainment
SQL3040\Administrator:Accessories\Entertainme
nt
SQL3040\Administrator
Administrative Tools
SQL3040\Administrator:Administrative Tools
SQL3040\Administrator
Startup SQL3040\Administrator:Startup
SQL3040\Administrator

[Startup Programs]
Program Command User Name Location
desktop desktop.ini NT AUTHORITY\SYSTEM
Startup
CTFMON.EXE c:\windows\system32\ctfmon.exe
NT AUTHORITY\SYSTEM HKU\S-1-5-
18\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
desktop desktop.ini SQL3040\Administrator
Startup
CTFMON.EXE c:\windows\system32\ctfmon.exe
SQL3040\Administrator HKU\S-1-5-21-
3778964393-3438175216-366707335-
500\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
desktop desktop.ini .DEFAULT Startup
CTFMON.EXE c:\windows\system32\ctfmon.exe
.DEFAULT
HKU\DEFAULT\SOFTWARE\Microsoft\Windows\Curre
ntVersion\Run
desktop desktop.ini All Users Common Startup

Service Manager
c:\program-1\micros-1\80\tools\bin\sqlmangr.e
xe /n All Users Common 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
WordPad Document "%programfiles%\windows
nt\accessories\wordpad.exe"
Windows Media Services DRM Storage object Not
Available
Bitmap Image mspaint.exe

[Windows Error Reporting]
Time Type Details
[Internet Settings]
[Internet Explorer]
[ Following are sub-categories of this main category ]
[Summary]
Item Value
Version 6.0.3790.0
Build 63790
Application Path C:\Program Files\Internet Explorer

Language English (United States)
Active Printer Not Available

Cipher Strength 128-bit
Content Advisor Disabled
IEAK Install No

[File Versions]
File Version Size Date Path
Company
actpxry.dll 6.0.3790.0 95 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
advpack.dll 6.0.3790.0 94 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
asctrls.ocx 6.0.3790.0 90 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
browsefc.dll 6.0.3790.0 62 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
browseui.dll 6.0.3790.0 1.033 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
cdfview.dll 6.0.3790.0 144 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
comctl32.dll 5.82.3790.0 561 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
dxtrans.dll 6.3.3790.0 198 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
dxtmsft.dll 6.3.3790.0 344 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
iecont.dll <File Missing> Not Available
Not Available
iecontlc.dll <File Missing> Not Available
Not Available
Available

```

```

iedkcs32.dll 16.0.3790.0 300 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
iepeers.dll 6.0.3790.0 230 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
iesetup.dll 6.0.3790.0 59 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
ieunit.inf Not Available 20 KB
25/3/2003 09:00:00 C:\WINDOWS\system32 Not
Available
iexplore.exe 6.0.3790.0 90 KB
25/3/2003 09:00:00 C:\Program Files\Internet
Explorer
imgutil.dll 5.2.3790.0 35 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
inetcp1.cpl 6.0.3790.0 303 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
inetcp1c.dll 6.0.3790.0 109 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
inseng.dll 6.0.3790.0 72 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
mlang.dll 6.0.3790.0 570 KB 25/3/2003
09:00:00 C:\WINDOWS\system32 Microsoft Corporation
msencode.dll 2002.10.4.0 112 KB
25/3/2003 09:00:00 C:\WINDOWS\system32 Not
Available
mshta.exe 6.0.3790.0 26 KB 25/3/2003
09:00:00 C:\WINDOWS\system32 Microsoft Corporation
mshtml.dll 6.0.3790.0 2,848 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
mshtml.tlb 6.0.3790.0 1,319 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
mshtml.ed.dll 6.0.3790.0 444 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
mshtmlr.dll 6.0.3790.0 55 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
msident.dll 6.0.3790.0 47 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
msidntld.dll 6.0.3790.0 15 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
msieftp.dll 6.0.3790.0 230 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
msrating.dll 6.0.3790.0 132 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
mstime.dll 6.0.3790.0 491 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
occache.dll 6.0.3790.0 89 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
proctexe.ocx 6.3.3790.0 78 KB
25/3/2003 09:00:00 C:\WINDOWS\system32 Intel
Corporation
sendmail.dll 6.0.3790.0 52 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation

```

```

shdoc1c.dll 6.0.3790.0 589 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
shdocvw.dll 6.0.3790.0 1,361 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
shfolder.dll 6.0.3790.0 23 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
shlwapi.dll 6.0.3790.0 281 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
tdc.ocx 1.3.0.3130 58 KB 25/3/2003
09:00:00 C:\WINDOWS\system32 Microsoft Corporation
url.dll 6.0.3790.0 36 KB 25/3/2003
09:00:00 C:\WINDOWS\system32 Microsoft Corporation
urlmon.dll 6.0.3790.0 502 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
webcheck.dll 6.0.3790.0 262 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation
wininet.dll 6.0.3790.0 609 KB
25/3/2003 09:00:00 C:\WINDOWS\system32
Microsoft Corporation

```

[Connectivity]

Item	Value
Connection Preference	Never dial

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\LocalService\Local Settings\Temporary Internet Files
Total Disk Space	Not Available
Available Disk Space	Not Available
Maximum Cache Size	Not Available
Available Cache Size	Not Available

[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
No personal certificate information available

[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
My Computer	Custom
Local intranet	Medium-low
Trusted sites	Medium
Internet High	
Restricted sites	High

IIS Registry Parameters

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\I
netInfo]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\I
netInfo\Parameters]
"ListenBackLog"=dword:00000019
"DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,
56,00,43,00,00,00,53,00,\
4d,00,54,00,50,00,53,00,56,00,43,00,00,00,00,00
"PoolThreadLimit"=dword:000007fe
"ThreadTimeout"=dword:00015180
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\I
netInfo\Performance]
"Library"="infoctr.dll"
"Open"="OpenINFOPerformanceData"
"Close"="CloseINFOPerformanceData"
"Collect"="CollectINFOPerformanceData"
"Last Counter"=dword:00000842
"Last Help"=dword:00000843
"First Counter"=dword:00000802
"First Help"=dword:00000803
"Library Validation
Code"=hex:50,25,f4,dd,ab,af,c4,01,10,25,00,00,00,00,00,
00
```

WWW Service Registry Parameters

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):43,00,3a,00,5c,00,57,00,49,00,4e,00,4e,00,54,00,5c,00,53,00,\
```

```
79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,69,00,6e,00,65,00,74,00,73,\
```

```
00,72,00,76,00,5c,00,69,00,6e,00,65,00,74,00,69,00,6e,00,66,00,6f,00,2e,00,\
65,00,78,00,65,00,00,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):49,00,49,00,53,00,41,00,44,00,4d,00,49,00,4e,00,00,00,\
00,00
"DependOnGroup"=hex(7):00,00
"ObjectName"="LocalSystem"
"Description"="Provides web connectivity and administration through the Internet Information Services snap-in."
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\ASP]
"NOTE"="This is for backward compatibility only."
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\ASP\Parameters]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Parameters]
"MajorVersion"=dword:00000005
"MinorVersion"=dword:00000000
"InstallPath"="C:\WINNT\System32\inet_srv"
"CertMapList"="C:\WINNT\System32\inet_srv\iisrmap.dll"
"AccessDeniedMessage"="Error: Access is Denied."
"Filter DLLs"=""
"LogFileDirectory"="C:\WINNT\System32\LogFiles"
"AcceptExOutstanding"=dword:00000028
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Parameters\ADCLaunch]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Parameters\ADCLaunch\AdvancedDataFactory]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Parameters\ADCLaunch\RDSServer.DataFactory]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Parameters\Script Map]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Parameters\Virtual Roots]
"/"="c:\inetpub\wwwroot,,205"
"/scripts"="c:\inetpub\scripts,,1"
"/IISHelp"="c:\winnt\help\iishe\p,,1"
"/IISAdmin"="c:\winnt\System32\inet_srv\iisadmin,,1"
"/IISamples"="c:\inetpub\iissamples,,1"
"/MSADC"="c:\program files\common files\system\msadc,,1"
"/vst_bin"="c:\program files\common files\microsoft shared\web server extensions\40\isapi,,1"
"/printers"="c:\winnt\web\printers,,201"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Performance]
"Library"="w3ctrs.dll"
"Open"="OpenW3PerformanceData"
```

```
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"Last Counter"=dword:000008e6
"Last Help"=dword:000008e7
"First Counter"=dword:00000844
"First Help"=dword:00000845
"Library Validation Code"=hex:aa,e1,58,e0,ab,af,c4,01,10,3d,00,00,00,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Security]
"Security"=hex:01,00,14,80,a0,00,00,00,ac,00,00,00,14,00,00,00,30,00,00,00,02,\
```

```
00,1c,00,01,00,00,00,02,80,14,00,ff,01,0f,00,01,01,00,00,00,00,01,00,00,\
```

```
00,00,02,00,70,00,04,00,00,00,00,00,18,00,fd,01,02,00,01,01,00,00,00,00,\
```

```
05,12,00,00,00,74,00,6f,00,00,00,1c,00,ff,01,0f,00,01,02,00,00,00,00,00,05,\
```

```
20,00,00,00,20,02,00,00,72,00,73,00,00,00,18,00,8d,01,02,00,01,01,00,00,00,\
```

```
00,00,05,0b,00,00,00,20,02,00,00,00,1c,00,fd,01,02,00,01,02,00,00,00,00,\
```

```
00,05,20,00,00,00,23,02,00,00,72,00,73,00,01,01,00,00,00,00,00,05,12,00,00,\
00,01,01,00,00,00,00,05,12,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\3SVC\Enum]
"0"="Root\LEGACY_W3SVC\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

COM+ Settings

```
TPCC.AllTxns:
  Activation:
    Enable Object Pooling selected
    Minimum Pool Size: 260
    Maximum Pool Size: 260
    Creation timeout: 60.000
    Enable Object Construction
    Enable Just in Time Activation
  Concurrency:
    Concurrency required
```

TPCC Application Registry Parameters

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\TPCC]
"Path"="c:\inetpub\wwwroot\"
"NumberOfDeliveryThreads"=dword:00000040
```

```
"MaxConnections"=dword:000080e8
"MaxPendingDeliveries"=dword:00000a8c
"DB_Protocol"="ODBC"
"TxnMonitor"="COM"
"DbServer"="sql3040"
"DbName"="tpcc"
"DbUser"="sa"
"DbPassword"=""
"COM_SinglePool"="YES"
```

Client System Configuration

System Information report written at: 06/22/2005 05:50:59 PM
[System Information]

[Following are sub-categories of this main category]

[System summary]

Item	Value
OS Name	Microsoft Windows 2000 Server
Version	5.0.2195 Service Pack 2 Build 2195
OS Manufacturer	Microsoft Corporation
System Name	CLI01
System Manufacturer	Intel
System Model	SBT2
System Type	x86-based PC
Processor	x86 Family 6 Model 8 Stepping 6 GenuineIntel ~1000 Mhz
Processor	x86 Family 6 Model 8 Stepping 6 GenuineIntel ~1000 Mhz
BIOS Version	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	CLI01\Administrator
Time Zone	E. South America Standard Time
Total Physical Memory	1,048,044 KB
Available Physical Memory	886,508 KB
Total Virtual Memory	3,570,192 KB
Available Virtual Memory	3,341,520 KB
Page File Space	2,522,148 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


```

C:\WINNT\System32\IR50_32.DLL R.5.10.15.2.55
737.50 KB (755,200 bytes) 12/7/1999
6:00:00 PM
c:\winnt\system32\iccvid.dll Radius Inc.
OK C:\WINNT\System32\ICCVID.DLL
1.10.0.6 108.00 KB (110,592 bytes)
12/7/1999 6:00:00 PM
c:\winnt\system32\msh263.driv Microsoft Corporation
OK
C:\WINNT\System32\MSH263.DRV 4.4.3385
252.27 KB (258,320 bytes) 1/31/2005
2:52:16 PM
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
6:00:00 PM
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
6:00:00 PM
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 6:00:00 PM

```

[CD-ROM]

```

Item Value
Drive D:
Description CD-ROM Drive
Media Loaded False
Media Type CD-ROM
Name LITEON CD-ROM LTNS26S
Manufacturer (Standard CD-ROM drives)
Status OK
Transfer Rate Not Available
SCSI Target ID 0
PNP Device ID IDE\CDROMLITEON_CD-
ROM_LTN526S_Y50G_\5&326853DD&0&0.
.0

```

[Sound Device]

```

Item Value
No sound devices

```

[Display]

```

Item Value
Name ATI Technologies Inc. 3D RAGE IIC PCI
PNP Device ID PCI\VEN_1002&DEV_4756&SUBSYS_00000000&REV_7A\
3&267A616A&0&10
Adapter Type ATI 3D RAGE IIC PCI (A21), ATI
Technologies Inc. compatible
Adapter Description ATI Technologies Inc. 3D RAGE IIC
PCI
Adapter RAM 4.00 MB (4,194,304 bytes)
Installed Drivers atiraged.dll
Driver Version 5.00.2174.1
INF File display.inf (atirage 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 00000416
PNP Device ID ACPI\PNP0303\4&23FD4C84&0
NumberOfFunctionKeys 12

```

[Pointing Device]

```

Item Value
Hardware Type PS/2 Compatible Mouse
Number of Buttons 5
Status OK
PNP Device ID ACPI\PNP0F13\4&23FD4C84&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] Intel 8255x-based PCI Ethernet
Adapter (10/100)
Adapter Type Ethernet 802.3
Product Name Intel 8255x-based PCI Ethernet
Adapter (10/100)
Installed True
PNP Device ID PCI\VEN_8086&DEV_1229&SUBSYS_12298086&REV_08\
3&267A616A&0&18
Last Reset 6/22/2005 9:14:40 AM
Index 0
Service Name E100B
IP Address 192.168.10.10
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:cb:78:9d
Service Name E100B
IRQ Number 18
I/O Port 0x1400-0x143F
Driver c:\winnt\system32\drivers\e100bnt5.sys
(85776, 4.02.38.0000)
Name [00000001] RAS Async Adapter
Adapter Type Not Available
Product Name RAS Async Adapter

```

```

Installed True
PNP Device ID Not Available
Last Reset 6/22/2005 9:14:40 AM
Index 1
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 [00000002] WAN Miniport (L2TP)
Adapter Type Not Available
Product Name WAN Miniport (L2TP)
Installed True
PNP Device ID ROOT\MS_L2TPMINIPOINT\0000
Last Reset 6/22/2005 9:14:40 AM
Index 2
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 [00000003] WAN Miniport (PPTP)
Adapter Type Wide Area Network (WAN)
Product Name WAN Miniport (PPTP)
Installed True
PNP Device ID ROOT\MS_PPTPMINIPOINT\0000
Last Reset 6/22/2005 9:14:40 AM
Index 3
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 [00000004] Direct Parallel
Adapter Type Not Available
Product Name Direct Parallel
Installed True
PNP Device ID ROOT\MS_PTMINIPOINT\0000
Last Reset 6/22/2005 9:14:40 AM
Index 4
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 [00000005] WAN Miniport (IP)
Adapter Type Not Available
Product Name WAN Miniport (IP)
Installed True
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 6/22/2005 9:14:40 AM
Index 5
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 (90096, 5.00.2195.2779)

Name [00000006] Intel(R) PRO/1000 T Server Adapter
Adapter Type Ethernet 802.3
Product Name Intel(R) PRO/1000 T Server Adapter
Installed True
PNP Device ID PCI\VEN_8086&DEV_1004&SUBSYS_10048086&REV_02\3&13C0B0C5&0&58
Last Reset 6/22/2005 9:14:40 AM
Index 6
Service Name E1000
IP Address 192.168.20.1
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 00:07:E9:0C:BD:53
Service Name E1000
IRQ Number 21
Driver c:\winnt\system32\drivers\e1000nt5.sys (126016, 7.0.37.0)

[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\NetBT_Tcpip_{878DA4C8-2B48-4A43-9F04-4633992B353B}]
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_{878DA4C8-2B48-4A43-9F04-4633992B353B}]
ConnectionlessService True
DATAGRAM 3

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_{56F5A191-3C54-47A8-A793-8848D94C0D83}]
ConnectionlessService SEQPACKET 0
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_{56F5A191-3C54-47A8-A793-8848D94C0D83}]
ConnectionlessService DATAGRAM 0
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_{CAA3118-1C8E-4E8E-8FF5-9168FA4C1628}]
ConnectionlessService SEQPACKET 1
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_{CAAA3118-1C8E-4E8E-8FF5-9168FA4C1628}] 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_{C36DA4A2-6791-4315-B705-941E5FAD05C7}] SEQUENTIAL 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_{C36DA4A2-6791-4315-B705-941E5FAD05C7}] 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 5.00.2195.2871
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 (62416, 5.00.2195.2780)
```

```
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 (62416, 5.00.2195.2780)
```

[Parallel]

```
Item Value
Name LPT1
PNP Device ID ACPI\PNP0400\4&23FD4C84&0
```

[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 17.08 GB (18,342,338,560 bytes)
Free Space 14.49 GB (15,559,356,416 bytes)
Volume Name
Volume Serial Number 2CE8316B
Partition Disk #0, Partition #0
Partition Size 17.08 GB (18,342,342,144 bytes)
Starting Offset 32256 bytes
Drive Description Disk drive
Drive Manufacturer (Standard disk drives)
Drive Model SEAGATE ST318406LC 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 18350599680 bytes
Drive TotalCylinders 2231
Drive TotalSectors 35841015
Drive TotalTracks 568905
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_00CF8086&REV_01\3&13C0B0C5&0&20
Device ID	PCI\VEN_9005&DEV_00CF&SUBSYS_00CF8086&REV_01\3&13C0B0C5&0&20
Device Map	Not Available
Index	Not Available
Max Number Controlled	Not Available
IRQ Number	16
I/O Port	0x1800-0x20FF
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_00CF8086&REV_01\3&13C0B0C5&0&21
Device ID	PCI\VEN_9005&DEV_00CF&SUBSYS_00CF8086&REV_01\3&13C0B0C5&0&21
Device Map	Not Available
Index	Not Available
Max Number Controlled	Not Available
IRQ Number	17
I/O Port	0x2000-0x20FF
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
No Problem Devices		

[USB]

Device	PNP Device ID
No USB Devices	

[Software Environment]

[Following are sub-categories of this main category]

[Drivers]

Name	Description	File	Type
Started	Start Mode	File	Type
Status	Error Control	State	Accept Pause
abiosdsk	Accept Stop		
abiosdsk	Not Available	Kernel Driver	
False	Disabled	Stopped	OK
Ignore	False	False	
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	Microsoft Embedded Controller Driver c:\winnt\system32\drivers\acpiec.sys Kernel Driver True Boot Running OK Normal False True
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 False Disabled Stopped OK Normal False False
aic116x	aic116x Not Available False Disabled Stopped OK Normal False False
aic78u2	aic78u2 Not Available False Disabled Stopped OK Normal False False
aic78xx	aic78xx Not Available False Disabled Stopped OK Normal False False
ami0nt	ami0nt Not Available False Disabled Stopped OK Normal False False
amsint	amsint Not Available False Disabled Stopped OK Normal False False
asc	asc Not Available False Disabled Stopped OK Normal False False
asc3350p	asc3350p Not Available False Disabled Stopped OK Normal False False
asc3550	asc3550 Not Available False Disabled Stopped OK Normal False False
asynmac	RAS Asynchronous Media Driver c:\winnt\system32\drivers\asynmac.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 False Disabled Stopped OK Ignore False False
atirage	atirage c:\winnt\system32\drivers\atiragem.sys Kernel Driver True Manual Running OK Ignore False True
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 False Disabled Stopped OK Normal False False

cd20xrnt	cd20xrnt Not Available 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 False System Stopped OK Ignore False False
cpqarray	Cpqarray Not Available False Disabled Stopped OK Normal False False
cpqarray2	cpqarray2 Not Available False Disabled Stopped OK Normal False False
cpqfca1m	cpqfca1m Not Available False Disabled Stopped OK Normal False False
cpqfws2e	cpqfws2e Not Available False Disabled Stopped OK Normal False False
dac960nt	dac960nt Not Available False Disabled Stopped OK Normal False False
deckzpsx	deckzpsx Not Available 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
dmbboot	dmbboot c:\winnt\system32\drivers\dmbboot.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 Adapter Driver c:\winnt\system32\drivers\e1000nt5.sys Kernel Driver True Manual 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
fd16_700	Normal	False	True	
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
fips	Fips	c:\winnt\system32\drivers\fips.sys	Kernel Driver	True
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
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	True	Manual	
Running	OK	Normal	False	True
ipsraidn	ipsraidn	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	
Normal	False	False		
isapnp	PnP ISA/EISA Bus Driver			
c:\winnt\system32\drivers\isapnp.sys	Kernel Driver	True	Boot	
Running	OK	Critical	False	True
kbdclass	Keyboard Class Driver			
c:\winnt\system32\drivers\kbdclass.sys	Kernel Driver	True	System	
Running	OK	Normal	False	True
ksecdd	KSecDD			
c:\winnt\system32\drivers\ksecdd.sys	Kernel Driver	True	Boot	
Running	OK	Normal	False	True

lbrtfdc	lbrtfdc	Not Available	Kernel Driver	
False	System	Stopped	OK	
Ignore	False	False		
lp6nds35	lp6nds35	Not Available	Kernel Driver	
False	Disabled	Stopped	OK	
Normal	False	False		
mmdd	mmdd	c:\winnt\system32\drivers\mmdd.sys	Kernel Driver	True
Running	OK	Ignore	False	True
modem	Modem	c:\winnt\system32\drivers\modem.sys	Kernel Driver	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
mqaac	MSMQ access control			
\\??c:\winnt\system32\drivers\mqaac.sys	Kernel Driver	True	Manual	
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	Running	OK	
System Driver	True	System	Running	OK
Normal	False	True		
msfs	Msfs	c:\winnt\system32\drivers\msfs.sys	File System Driver	True
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
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	Running	OK	
System Driver	True	System	Running	OK
Normal	False	True		

netbt	NetBios over Tcpip			
c:\winnt\system32\drivers\netbt.sys	Kernel Driver	True	System	
Running	OK	Normal	False	True
netdetect	NetDetect			
c:\winnt\system32\drivers\netdetect.sys	Kernel Driver	False	Manual	
Stopped	OK	Normal	False	False
npfs	Npfs	c:\winnt\system32\drivers\npfs.sys	File System Driver	True
Running	OK	Normal	False	True
ntfs	Ntfs	c:\winnt\system32\drivers\ntfs.sys	File System Driver	True
Running	OK	Normal	Disabled	True
null	Null	c:\winnt\system32\drivers\null.sys	Kernel Driver	True
Running	OK	Normal	System	True
nwlkflt	IPX Traffic Filter Driver			
c:\winnt\system32\drivers\nwlkflt.sys	Kernel Driver	False	Manual	
Stopped	OK	Normal	False	False
nwlkfw	IPX Traffic Forwarder Driver			
c:\winnt\system32\drivers\nwlkfw.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			
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
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\rasppptp.sys	Kernel Driver	True	Manual	
Running	OK	Normal	False	True

```

ptlink Direct Parallel Link Driver
c:\winnt\system32\drivers\ptlink.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
ras12tp WAN Miniport (L2TP)
c:\winnt\system32\drivers\ras12tp.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
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
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
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\os2dll;

```

```

Path
%SystemRoot%\system32;%SystemRoot%;%SystemRo
t%\system32\wbem <SYSTEM>
windir %SystemRoot% <SYSTEM>
OS windows_NT <SYSTEM>
PROCESSOR_ARCHITECTURE x86 <SYSTEM>
PROCESSOR_LEVEL 6 <SYSTEM>
PROCESSOR_IDENTIFIER x86 Family 6 Model 8
Stepping 6, GenuineIntel <SYSTEM>
PROCESSOR_REVISION 0806 <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
CLIO1\Administrator
TEMP %USERPROFILE%\Local Settings\Temp
CLIO1\Administrator

[Jobs]
[ Following are sub-categories of this main category ]
[Print]
Document Size Owner Notify Status Time
Submitted Start Time Until Time
ID Elapsed Time Pages Printed Job
Priority Parameters Driver Name
Print Processor Host Print Queue Data
Type Name
No print jobs

[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
system Not Available 8 8 0
1413120 Not Available Unknown
Unknown Unknown
smss.exe c:\winnt\system32\smss.exe 164 11
204800 1413120 6/22/2005 12:15:00 PM
5.00.2195.2901 44.27 KB (45,328 bytes)
5/4/2001 12:05:02 PM
csrss.exe Not Available 188 13 Not
Available Not Available 6/22/2005 12:15:03 PM
Unknown Unknown
winlogon.exe c:\winnt\system32\winlogon.exe
184 13 204800 1413120
6/22/2005 12:15:04 PM 5.00.2195.2953
173.77 KB (177,936 bytes) 5/4/2001
12:05:02 PM
services.exe c:\winnt\system32\services.exe
236 9 204800 1413120
6/22/2005 12:15:05 PM 5.00.2195.2780

```

```

12:05:02 PM 86.77 KB (88,848 bytes) 5/4/2001
lsass.exe c:\winnt\system32\lsass.exe 248 9
204800 1413120 6/22/2005 12:15:05 PM
5.00.2195.2964 32.77 KB (33,552 bytes)
svchost.exe c:\winnt\system32\svchost.exe 416
8 204800 1413120 6/22/2005
12:15:08 PM 5.00.2134.1 7.77 KB (7,952
bytes) 12/7/1999 6:00:00 PM
spoolsv.exe c:\winnt\system32\spoolsv.exe 444
204800 1413120 6/22/2005
12:15:08 PM 5.00.2161.1 43.77 KB
(44,816 bytes) 1/31/2005 12:45:44 PM
msdtc.exe c:\winnt\system32\msdtc.exe 472 8
204800 1413120 6/22/2005 12:15:09 PM
1999.9.3421.3 6.77 KB (6,928 bytes)
svchost.exe c:\winnt\system32\svchost.exe 596
8 204800 1413120 6/22/2005
12:15:09 PM 5.00.2134.1 7.77 KB (7,952
bytes) 12/7/1999 6:00:00 PM
llssrv.exe c:\winnt\system32\llssrv.exe 620
204800 1413120 6/22/2005
12:15:10 PM 5.00.2195.2649 114.27 KB
(117,008 bytes) 5/4/2001 12:05:02 PM
regsvc.exe c:\winnt\system32\regsvc.exe 672
204800 1413120 6/22/2005
12:15:10 PM 5.00.2195.2104 65.27 KB
(66,832 bytes) 5/4/2001 12:05:02 PM
mstask.exe c:\winnt\system32\mstask.exe 716
204800 1413120 6/22/2005
12:15:10 PM 4.71.2195.1 115.27 KB
(118,032 bytes) 1/31/2005 2:52:22 PM
mqsvcs.exe c:\winnt\system32\mqsvcs.exe 848 8
204800 1413120 6/22/2005 12:15:14 PM
5.00.0720 13.77 KB (14,096 bytes)
1/31/2005 12:50:01 PM
dfssvc.exe c:\winnt\system32\dfssvc.exe 1080
204800 1413120 6/22/2005
12:15:19 PM 5.00.2195.2841 88.27 KB
(90,384 bytes) 5/4/2001 12:05:02 PM
explorer.exe c:\winnt\explorer.exe 272
204800 1413120 6/22/2005
12:15:20 PM 5.00.3315.2846 237.27 KB
(242,960 bytes) 5/4/2001 12:05:02 PM
internat.exe c:\winnt\system32\internat.exe
1180 204800 1413120
6/22/2005 12:15:20 PM 5.00.2920.0000
20.27 KB (20,752 bytes) 12/7/1999
6:00:00 PM
svchost.exe c:\winnt\system32\svchost.exe 1300
8 204800 1413120 6/22/2005
12:15:31 PM 5.00.2134.1 7.77 KB (7,952
bytes) 12/7/1999 6:00:00 PM
dllhost.exe Not Available 1312 8
Not Available Not Available
6/22/2005 1:00:01 PM Unknown
Unknown Unknown
mmc.exe c:\winnt\system32\mmc.exe 2616 8
204800 1413120 6/22/2005 5:49:22 PM
5.00.2195.2301 589.27 KB (603,408 bytes)
5/4/2001 12:05:02 PM
winmgmt.exe c:\winnt\system32\wbem\winmgmt.exe
2584 8 204800 1413120
6/22/2005 5:49:23 PM 1.50.1085.0029
192.08 KB (196,685 bytes) 5/4/2001
12:05:02 PM
inetinfo.exe c:\winnt\system32\inetinfo.exe 1656
8 204800 1413120 6/22/2005

```

```

5:49:41 PM 5.00.0984 14.27 KB (14,608 bytes)
1/31/2005 12:49:51 PM
rsvp.exe c:\winnt\system32\rsvp.exe 1256 8
204800 1413120 6/22/2005 5:50:44 PM
5.00.2167.1 172.77 KB (176,912 bytes)
12/7/1999 6:00:00 PM

```

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer
Path				
traffic.dll	5.00.2139.1	30.77 KB	12/7/1999 6:00:00 PM	Microsoft Corporation
(31,504 bytes)				
rsvp.exe	5.00.2167.1	172.77 KB (176,912 bytes)	12/7/1999 6:00:00 PM	Microsoft Corporation
svcxext.dll	5.00.0984 39.77 KB (40,720 bytes)		1/31/2005 12:49:52 PM	Microsoft Corporation
admexs.dll	5.00.0984 27.77 KB (28,432 bytes)		1/31/2005 12:49:51 PM	Microsoft Corporation
wamreg.dll	5.00.0984 45.77 KB (46,864 bytes)		1/31/2005 12:49:56 PM	Microsoft Corporation
metadata.dll	5.00.0984 68.77 KB (70,416 bytes)		1/31/2005 12:49:51 PM	Microsoft Corporation
iismap.dll	5.00.0984 55.77 KB (57,104 bytes)		1/31/2005 12:49:52 PM	Microsoft Corporation
nsepm.dll	5.00.0984 43.27 KB (44,304 bytes)		1/31/2005 12:49:52 PM	Microsoft Corporation
admwxprox.dll	5.00.0984 31.77 KB (32,528 bytes)		1/31/2005 12:50:05 PM	Microsoft Corporation
coadmin.dll	5.00.0984 39.27 KB (40,208 bytes)		1/31/2005 12:49:53 PM	Microsoft Corporation
iisadmin.dll	5.00.0984 15.27 KB (15,632 bytes)		1/31/2005 12:49:51 PM	Microsoft Corporation
rpcref.dll	5.00.0984 4.27 KB (4,368 bytes)		1/31/2005 12:49:52 PM	Microsoft Corporation
iisrtrl.dll	5.00.0984 119.77 KB (122,640 bytes)		1/31/2005 12:49:52 PM	Microsoft Corporation
inetinfo.exe	5.00.0984 14.27 KB (14,608 bytes)		1/31/2005 12:49:51 PM	Microsoft Corporation
wshnetbs.dll	5.00.2134.1 7.77 KB (7,952 bytes)		12/7/1999 6:00:00 PM	Microsoft Corporation
ntevt.dll	1.50.1085.0000 192.06 KB (196,669 bytes)		12/7/1999 6:00:00 PM	Microsoft Corporation
perfos.dll	5.00.2155.1 21.27 KB (21,776 bytes)		12/7/1999 6:00:00 PM	Microsoft Corporation

```

psapi.dll 5.00.2134.1 28.27 KB (28,944 bytes)
12/7/1999 6:00:00 PM Microsoft Corporation
framedyn.dll c:\winnt\system32\psapi.dll
1.50.1085.0000 164.05 KB (167,992 bytes)
12/7/1999 6:00:00 PM Microsoft Corporation
cimwin32.dll c:\winnt\system32\wbem\framedyn.dll
1.50.1085.0038 1.02 MB (1,073,232 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
adslp.dll c:\winnt\system32\wbem\cimwin32.dll
5.00.2195.2778 119.77 KB (122,640 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
provthrd.dll c:\winnt\system32\adslp.dll
1.50.1085.0000 68.07 KB (69,708 bytes)
1/31/2005 2:52:32 PM Microsoft Corporation
dsprov.dll c:\winnt\system32\wbem\provthrd.dll
1.50.1085.0000 196.06 KB (200,761 bytes)
1/31/2005 2:52:32 PM Microsoft Corporation
mofd.dll c:\winnt\system32\wbem\dsprov.dll
1.50.1085.0007 136.07 KB (139,332 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
wmiprovd.dll c:\winnt\system32\wbem\mofd.dll
1.50.1085.0032 108.07 KB (110,660 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
wbemess.dll c:\winnt\system32\wbem\wmiprovd.dll
1.50.1085.0039 364.07 KB (372,804 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
wbemcore.dll c:\winnt\system32\wbem\wbemess.dll
1.50.1085.0036 628.07 KB (643,140 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
winmgmt.exe c:\winnt\system32\wbem\wbemcore.dll
1.50.1085.0029 192.08 KB (196,685 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
fastprox.dll c:\winnt\system32\wbem\winmgmt.exe
1.50.1085.0037 144.08 KB (147,536 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
wbemsvc.dll c:\winnt\system32\wbem\fastprox.dll
1.50.1085.0007 40.07 KB (41,036 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
wbemcomn.dll c:\winnt\system32\wbem\wbemsvc.dll
1.50.1085.0021 692.07 KB (708,675 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
wbemprox.dll c:\winnt\system32\wbem\wbemcomn.dll
1.50.1085.0045 40.08 KB (41,040 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
cabinet.dll c:\winnt\system32\wbem\wbemprox.dll
5.00.2147.1 54.77 KB (56,080 bytes)
12/7/1999 6:00:00 PM Microsoft Corporation
msinfo32.dll c:\winnt\system32\cabinet.dll
5.00.2177.1 312.27 KB (319,760 bytes)
1/31/2005 2:52:36 PM Microsoft Corporation
files\common_files\microsoft_shared\msinfo32\msinfo32.dll
5.00.3103.1000 236.77 KB (242,448 bytes)
5/4/2001 12:05:02 PM Microsoft Corporation
mmcndmgr.dll c:\winnt\system32\cmdlg32.dll
5.00.2178.1 815.27 KB (834,832 bytes)
12/7/1999 6:00:00 PM

```



```

Microsoft Corporation
c:\winnt\system32\mcmdmgr.dll
mmc.exe 5.00.2195.2301 589.27 KB (603,408 bytes)
5/4/2001 12:05:02 PM Microsoft
Corporation c:\winnt\system32\mmc.exe
h323.tsp 5.00.2195.2283 248.77 KB (254,736 bytes)
5/4/2001 12:05:02 PM Microsoft
Corporation c:\winnt\system32\h323.tsp
ipconf.tsp 5.00.2143.1 10.77 KB
(11,024 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\ipconf.tsp
ndptsp.tsp 5.00.2143.1 38.27 KB
(39,184 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\ndptsp.tsp
kmdsp.tsp 5.00.2150.1 17.77 KB
(18,192 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\kmdsp.tsp
ntmarta.dll 5.00.2195.2862 98.77 KB
(101,136 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\ntmarta.dll
uniplat.dll 5.00.2151.1 13.77 KB
(14,096 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\uniplat.dll
unimdm.tsp 5.00.2175.1 196.77 KB
(201,488 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\unimdm.tsp
tapisrv.dll 5.00.2195.2955 169.27 KB
(173,328 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\tapisrv.dll
internat.exe 5.00.2920.0000 20.27 KB
(20,752 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\internat.exe
shdoclc.dll 5.00.3315.2879 324.50 KB
(332,288 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\shdoclc.dll
faxshell.dll 5.00.2134.1 8.27 KB (8,464
bytes) 12/7/1999 6:00:00 PM Microsoft
Corporation c:\winnt\system32\faxshell.dll
msacm32.dll 5.00.2134.1 65.27 KB
(66,832 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\msacm32.dll
avifil32.dll 5.00.2134.1 76.27 KB
(78,096 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\avifil32.dll
msvfw32.dll 5.00.2134.1 113.77 KB
(116,496 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\msvfw32.dll
docprop2.dll 5.00.2178.1 297.77 KB
(304,912 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\docprop2.dll
wininet.dll 5.00.3315.1000 456.77 KB
(467,728 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\wininet.dll
msdbg.dll 6.00.8424 67.50 KB (69,120 bytes)
1/31/2005 12:51:14 PM Microsoft
Corporation c:\winnt\system32\msdbg.dll

```

```

pdm.dll 6.00.8424 179.27 KB (183,574 bytes)
1/31/2005 12:51:14 PM Microsoft
Corporation c:\winnt\system32\pdm.dll
mlang.dll 5.00.3103.1000 510.77 KB (523,024 bytes)
5/4/2001 12:05:02 PM Microsoft
Corporation c:\winnt\system32\mlang.dll
imm32.dll 5.00.2195.2821 94.27 KB (96,528 bytes)
5/4/2001 12:05:02 PM Microsoft
Corporation c:\winnt\system32\imm32.dll
indic11.dll 5.00.2920.0000 11.27 KB
(11,536 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\indic11.dll
urlmon.dll 5.00.3315.1000 441.27 KB
(451,856 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\urlmon.dll
linkinfo.dll 5.00.2134.1 15.77 KB
(16,144 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\linkinfo.dll
browserc.dll 5.00.3315.2846 34.50 KB
(35,328 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\browserc.dll
powrprof.dll 5.00.3103.1000 13.27 KB
(13,584 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\powrprof.dll
batmeter.dll 5.00.3103.1000 20.27 KB
(20,752 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\batmeter.dll
stobject.dll 5.00.2195.2780 79.27 KB
(81,168 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\stobject.dll
msi.dll 1.11.2405.0 1.69 MB (1,767,184 bytes)
5/4/2001 12:05:02 PM Microsoft
Corporation c:\winnt\system32\msi.dll
webcheck.dll 5.00.3315.1000 251.77 KB
(257,808 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\webcheck.dll
netui1.dll 5.00.2134.1 210.27 KB
(215,312 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\netui1.dll
netui0.dll 5.00.2134.1 70.27 KB
(71,952 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\netui0.dll
ntlanman.dll 5.00.2157.1 35.27 KB
(36,112 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\ntlanman.dll
mydocs.dll 5.00.2920.0000 55.77 KB
(57,104 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\mydocs.dll
ntshrui.dll 5.00.2134.1 46.77 KB
(47,888 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\ntshrui.dll
browseui.dll 5.00.3315.2846 788.77 KB
(807,696 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\browseui.dll
shdocvw.dll 5.00.3315.2879 1.05 MB
(1,104,144 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\shdocvw.dll

```

```

explorer.exe 5.00.3315.2846 237.27 KB
(242,960 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\explorer.exe
dfssvc.exe 5.00.2195.2841 88.27 KB
(90,384 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\dfssvc.exe
mqdscli.dll 5.00.0720 73.77 KB (75,536 bytes)
1/31/2005 12:50:01 PM Microsoft
Corporation c:\winnt\system32\mqdscli.dll
mqlogmgr.dll 1999.8.3413.7 85.27 KB
(87,312 bytes) 1/31/2005 12:50:19 PM
Microsoft Corporation
c:\winnt\system32\mqlogmgr.dll
mqutil.dll 5.00.0720 105.27 KB (107,792 bytes)
1/31/2005 12:50:01 PM Microsoft
Corporation c:\winnt\system32\mqutil.dll
security.dll 5.00.2154.1 5.77 KB (5,904
bytes) 12/7/1999 6:00:00 PM Microsoft
Corporation c:\winnt\system32\security.dll
mqsec.dll 5.00.0720 68.77 KB (70,416 bytes)
1/31/2005 12:50:01 PM Microsoft
Corporation c:\winnt\system32\mqsec.dll
mqqm.dll 5.00.0721 404.27 KB (413,968 bytes)
1/31/2005 12:50:01 PM Microsoft
Corporation c:\winnt\system32\mqqm.dll
mqsvc.exe 5.00.0720 13.77 KB (14,096 bytes)
1/31/2005 12:50:01 PM Microsoft
Corporation c:\winnt\system32\mqsvc.exe
msidle.dll 5.00.2920.0000 6.27 KB (6,416
bytes) 12/7/1999 6:00:00 PM Microsoft
Corporation c:\winnt\system32\msidle.dll
mstask.exe 4.71.2195.1 115.27 KB
(118,032 bytes) 1/31/2005 2:52:22 PM
Microsoft Corporation
c:\winnt\system32\mstask.exe
regsvcs.exe 5.00.2195.2104 65.27 KB
(66,832 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\regsvcs.exe
llsrpc.dll 5.00.2149.1 45.77 KB
(46,864 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\llsrpc.dll
llssrv.exe 5.00.2195.2649 114.27 KB
(117,008 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\llssrv.exe
ntmsdba.dll 5.00.2195.2779 167.27 KB
(171,280 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\ntmsdba.dll
ipbootp.dll 5.00.2168.1 33.77 KB
(34,576 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\ipbootp.dll
rastls.dll 5.00.2195.2671 47.77 KB
(48,912 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\rastls.dll
raschap.dll 5.00.2195.2671 34.77 KB
(35,600 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\raschap.dll
rasppp.dll 5.00.2195.2671 192.77 KB
(197,392 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\rasppp.dll
rastapi.dll 5.00.2195.2671 52.77 KB
(54,032 bytes) 5/4/2001 12:05:02 PM

```

```

Microsoft Corporation
c:\winnt\system32\rastapi.dll
rasdlg.dll 5.00.2195.2671 514.27 KB
(526,608 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\rasdlg.dll
netcfgx.dll 5.00.2195.2228 534.77 KB
(547,600 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\netcfgx.dll
rasmans.dll 5.00.2195.2728 147.27 KB
(150,800 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\rasmans.dll
wmi.dll 5.00.2191.1 6.27 KB (6,416 bytes)
12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\wmi.dll
netshell.dll 5.00.2195.2779 457.27 KB
(468,240 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\netshell.dll
netman.dll 5.00.2195.2779 89.27 KB
(91,408 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\netman.dll
sens.dll 5.00.2163.1 36.77 KB (37,648 bytes)
12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\sens.dll
ntmssvc.dll 5.00.2195.2779 391.27 KB
(400,656 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\ntmssvc.dll
es.dll 2000.2.3471.1 222.27 KB (227,600 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\es.dll
mtxoci.dll 2000.2.3471.1 101.77 KB
(104,208 bytes) 1/31/2005 12:49:59 PM
Microsoft Corporation
c:\winnt\system32\mtxoci.dll
resutils.dll 5.00.2195.2787 39.77 KB
(40,720 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\resutils.dll
clusapi.dll 5.00.2195.2104 54.27 KB
(55,568 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\clusapi.dll
msvcp50.dll 5.00.2051.552.50 KB (565,760 bytes)
12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\msvcp50.dll
xolehlp.dll 1999.9.3421.3 17.27 KB
(17,680 bytes) 1/31/2005 12:50:10 PM
Microsoft Corporation
c:\winnt\system32\xolehlp.dll
msdtclog.dll 1999.9.3421.3 89.77 KB
(91,920 bytes) 1/31/2005 12:50:10 PM
Microsoft Corporation
c:\winnt\system32\msdtclog.dll
mtxclu.dll 2000.2.3471.1 51.27 KB
(52,496 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\mtxclu.dll
msdtcprx.dll 2000.2.3471.1 665.77 KB
(681,744 bytes) 1/31/2005 12:49:59 PM
Microsoft Corporation
c:\winnt\system32\msdtcprx.dll
txfaux.dll 2000.2.3471.1 374.27 KB
(383,248 bytes) 1/31/2005 12:49:59 PM
Microsoft Corporation
c:\winnt\system32\txfaux.dll
msdtctm.dll 2000.2.3471.1 1.07 MB
(1,120,528 bytes) 1/31/2005 12:49:59 PM

```

```

Microsoft Corporation
c:\winnt\system32\msdtctm.dll
msdtc.exe 1999.9.3421.3 6.77 KB (6,928 bytes)
1/31/2005 12:50:10 PM
Microsoft Corporation
c:\winnt\system32\msdtc.exe
inetpp.dll 5.00.2195.2842 65.27 KB
(66,832 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\inetpp.dll
win32spl.dll 5.00.2195.2780 92.27 KB
(94,480 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\win32spl.dll
usbmon.dll 5.00.2195.2780 11.27 KB
(11,536 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\usbmon.dll
tcpmon.dll 5.00.2195.2780 40.77 KB
(41,744 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\tcpmon.dll
pjlmon.dll 5.00.2165.1 12.77 KB
(13,072 bytes) 11/30/1999 9:39:36 PM
Microsoft Corporation
c:\winnt\system32\pjlmon.dll
cnbjmon.dll 5.00.2134.1 43.77 KB
(44,816 bytes) 11/30/1999 9:38:48 PM
Microsoft Corporation
c:\winnt\system32\cnbjmon.dll
localspl.dll 5.00.2195.2793 246.77 KB
(252,688 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\localspl.dll
spoolss.dll 5.00.2161.1 61.77 KB
(63,248 bytes) 1/31/2005 12:45:44 PM
Microsoft Corporation
c:\winnt\system32\spoolss.dll
spoolsv.exe 5.00.2161.1 43.77 KB
(44,816 bytes) 1/31/2005 12:45:44 PM
Microsoft Corporation
c:\winnt\system32\spoolsv.exe
rpcss.dll 5.00.2195.2815 231.27 KB (236,816 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\rpcss.dll
svchost.exe 5.00.2134.1 7.77 KB (7,952 bytes)
12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\svchost.exe
iissuba.dll 5.00.0984.9.77 KB (10,000 bytes)
12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\iissuba.dll
dssenh.dll 5.00.2195.2228 142.77 KB
(146,192 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\dssenh.dll
oakley.dll 5.00.2195.2785 378.77 KB
(387,856 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\oakley.dll
mfc42u.dll 6.00.8665.0 972.05 KB
(995,384 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\mfc42u.dll
polagent.dll 5.00.2183.1 108.27 KB
(110,864 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\polagent.dll
scecli.dll 5.00.2195.2780 105.27 KB
(107,792 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\scecli.dll

```

```

atld.dll 3.00.8449 57.56 KB (58,938 bytes)
12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\atld.dll
certcli.dll 5.00.2195.2778 130.77 KB
(133,904 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\certcli.dll
esent.dll 6.0.3940.13 1.08 MB (1,135,376 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\esent.dll
ntdsatq.dll 5.00.2195.2878 31.27 KB
(32,016 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\ntdsatq.dll
ntdsa.dll 5.00.2195.2899 990.77 KB (1,014,544 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\ntdsa.dll
kdcsvc.dll 5.00.2195.2878 137.77 KB
(141,072 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\kdcsvc.dll
sfmapi.dll 5.00.2134.1 38.77 KB
(39,696 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\sfmapi.dll
rassfm.dll 5.00.2195.2671 21.27 KB
(21,776 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\rassfm.dll
mpr.dll 5.00.2195.2779 53.27 KB (54,544 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\mpr.dll
rsabase.dll 5.00.2195.2228 128.27 KB
(131,344 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\rsabase.dll
schannel.dll 5.00.2195.2922 138.27 KB
(141,584 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\schannel.dll
netlogon.dll 5.00.2195.2865 357.77 KB
(366,352 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\netlogon.dll
kerberos.dll 5.00.2195.2913 198.77 KB
(203,536 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\kerberos.dll
msprivs.dll 5.00.2154.1 41.50 KB
(42,496 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\msprivs.dll
samsrv.dll 5.00.2195.2918 369.77 KB
(378,640 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\samsrv.dll
lsasrv.dll 5.00.2195.2964 492.77 KB
(504,592 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\lsasrv.dll
lsass.exe 5.00.2195.2964 32.77 KB (33,552 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\lsass.exe
ntlsapi.dll 5.00.2134.1 6.77 KB (6,928 bytes)
12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\ntlsapi.dll
xactsrv.dll 5.00.2134.1 90.27 KB
(92,432 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\xactsrv.dll
wmicore.dll 5.00.2195.2842 72.27 KB
(74,000 bytes) 5/4/2001 12:05:02 PM

```

Microsoft Corporation
 c:\winnt\system32\wmi\core.dll
 rasadhlp.dll 5.00.2168.1 7.27 KB (7,440 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\rasadhlp.dll
 winnr.dll 5.00.2160.1 18.77 KB (19,216 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\winnr.dll
 rnr20.dll 5.00.2195.2871 35.77 KB (36,624 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\rnr20.dll
 wshtcpip.dll 5.00.2195.2104 17.27 KB (17,680 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\wshtcpip.dll
 msafd.dll 5.00.2195.2779 106.77 KB (109,328 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\msafd.dll
 mssock.dll 5.00.2195.2871 62.77 KB (64,272 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\mssock.dll
 msgsvc.dll 5.00.2195.2939 34.27 KB (35,088 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\msgsvc.dll
 alrsvc.dll 5.00.2134.1 17.77 KB (18,192 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\alrsvc.dll
 browser.dll 5.00.2195.2778 48.27 KB (49,424 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\browser.dll
 trkws.dll 5.00.2166.1 88.77 KB (90,896 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\trkws.dll
 seclogon.dll 5.00.2135.1 15.77 KB (16,144 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\seclogon.dll
 psbase.dll 5.00.2195.2779 111.77 KB (114,448 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\psbase.dll
 cryptsvc.dll 5.00.2181.1 61.77 KB (63,248 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\cryptsvc.dll
 cryptdll.dll 5.00.2135.1 41.27 KB (42,256 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\cryptdll.dll
 wkssvc.dll 5.00.2195.2780 95.27 KB (97,552 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\wkssvc.dll
 srsvsvc.dll 5.00.2195.2904 79.27 KB (81,168 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\srsvsvc.dll
 cfgmgr32.dll 5.00.2134.1 16.77 KB (17,168 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\cfgmgr32.dll
 dmserver.dll 2195.2778.297.3 11.77 KB (12,048 bytes) 5/4/2001 12:05:02 PM VERITAS Software Corp.
 c:\winnt\system32\dmserver.dll

winsta.dll 5.00.2195.2386 36.77 KB (37,648 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\winsta.dll
 lmhsvc.dll 5.00.2195.2778 9.77 KB (10,000 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\lmhsvc.dll
 dnrsrslvr.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\dnrsrslvr.dll
 clbcatq.dll 2000.2.3471.1 496.77 KB (508,688 bytes) 1/31/2005 12:49:57 PM Microsoft Corporation
 c:\winnt\system32\clbcatq.dll
 tapi32.dll 5.00.2182.1 123.27 KB (126,224 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\tapi32.dll
 rasman.dll 5.00.2195.2780 54.77 KB (56,080 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\rasman.dll
 rasapi32.dll 5.00.2195.2671 189.77 KB (194,320 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\rasapi32.dll
 rtutils.dll 5.00.2168.1 43.77 KB (44,816 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\rtutils.dll
 adslrpc.dll 5.00.2195.2842 127.27 KB (130,320 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\adslrpc.dll
 activeds.dll 5.00.2195.2778 174.77 KB (178,960 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\activeds.dll
 oleaut32.dll 2.40.4517.612.27 KB (626,960 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\oleaut32.dll
 mprapi.dll 5.00.2181.1 79.27 KB (81,168 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\mprapi.dll
 iphlpapi.dll 5.00.2173.2 67.77 KB (69,392 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\iphlpapi.dll
 icmp.dll 5.00.2134.1 7.27 KB (7,440 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\icmp.dll
 dhcpcsvc.dll 5.00.2195.2778 88.77 KB (90,896 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\dhcpcsvc.dll
 eventlog.dll 5.00.2178.1 43.77 KB (44,816 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\eventlog.dll
 ntdsapi.dll 5.00.2195.2661 55.77 KB (57,104 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\ntdsapi.dll
 scesrv.dll 5.00.2195.2780 226.27 KB (231,696 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\scesrv.dll
 umpnmgm.dll 5.00.2182.1 86.27 KB (88,336 bytes) 12/7/1999 6:00:00 PM

Microsoft Corporation
 c:\winnt\system32\umpnmgm.dll
 services.exe 5.00.2195.2780 86.77 KB (88,848 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\services.exe
 msv1_0.dll 5.00.2195.2900 111.77 KB (114,448 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\msv1_0.dll
 csui.dll 5.00.2195.2959 228.27 KB (233,744 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\csui.dll
 winspool.drv 5.00.2195.2780 109.77 KB (112,400 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\winspool.drv
 wincard.dll 5.00.2134.1 77.27 KB (79,120 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\wincard.dll
 wlnotify.dll 5.00.2195.2780 53.77 KB (55,056 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\wlnotify.dll
 csddl.dll 5.00.2195.2401 98.27 KB (100,624 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\csddl.dll
 lz32.dll 5.00.2134.1 9.77 KB (10,000 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\lz32.dll
 version.dll 5.00.2134.1 15.77 KB (16,144 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\version.dll
 rsaenh.dll 5.00.2195.2228 130.77 KB (133,904 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\rsaenh.dll
 mscat32.dll 5.131.2134.1 7.77 KB (7,952 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\mscat32.dll
 ole32.dll 5.00.2195.2887 969.77 KB (993,040 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\ole32.dll
 imagehlp.dll 5.00.2195.2778 125.77 KB (128,784 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\imagehlp.dll
 msasn1.dll 5.00.2134.1 51.27 KB (52,496 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\msasn1.dll
 crypt32.dll 5.131.2195.2833 451.27 KB (462,096 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\crypt32.dll
 wintrust.dll 5.131.2195.2779 162.27 KB (166,160 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\wintrust.dll
 setupapi.dll 5.00.2195.2663 555.77 KB (569,104 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\setupapi.dll
 winmm.dll 5.00.2161.1 184.77 KB (189,200 bytes) 12/7/1999 6:00:00 PM Microsoft Corporation
 c:\winnt\system32\winmm.dll
 comctl32.dll 5.81 537.77 KB (550,672 bytes) 5/4/2001 12:05:02 PM Microsoft Corporation
 c:\winnt\system32\comctl32.dll

```

shlwapi.dll 5.00.3315.1000 282.77 KB
(289,552 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\shlwapi.dll
shell32.dll 5.00.3315.2902 2.25 MB
(2,359,056 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\shell32.dll
msgina.dll 5.00.2195.2779 324.27 KB
(332,048 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\msgina.dll
wsock32.dll 5.00.2195.2871 21.27 KB
(21,776 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\wsock32.dll
dnsapi.dll 5.00.2195.2785 130.77 KB
(133,904 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\dnsapi.dll
wldap32.dll 5.00.2195.2797 125.27 KB
(128,272 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\wldap32.dll
ws2help.dll 5.00.2134.1 17.77 KB
(18,192 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\ws2help.dll
ws2_32.dll 5.00.2195.2780 67.77 KB
(69,392 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\ws2_32.dll
samlib.dll 5.00.2195.2780 49.77 KB
(50,960 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\samlib.dll
netrap.dll 5.00.2134.1 11.27 KB
(11,536 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\netrap.dll
netapi32.dll 5.00.2195.2808 303.77 KB
(311,056 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\netapi32.dll
profmap.dll 5.00.2181.1 29.27 KB
(29,968 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\profmap.dll
secur32.dll 5.00.2195.2862 46.77 KB
(47,888 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\secur32.dll
sfc.dll 5.00.2195.2896 92.11 KB (94,320 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\sfc.dll
nddeapi.dll 5.00.2137.1 15.27 KB
(15,632 bytes) 12/7/1999 6:00:00 PM
Microsoft Corporation
c:\winnt\system32\nddeapi.dll
userenv.dll 5.00.2195.2780 361.77 KB
(370,448 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\userenv.dll
user32.dll 5.00.2195.2821 392.77 KB
(402,192 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\user32.dll
gdi32.dll 5.00.2195.2778 228.77 KB (234,256 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\gdi32.dll
rpcrt4.dll 5.00.2195.2832 437.27 KB
(447,760 bytes) 5/4/2001 12:05:02 PM

```

```

Microsoft Corporation
c:\winnt\system32\rpcrt4.dll
advapi32.dll 5.00.2195.2867 351.77 KB
(360,208 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\advapi32.dll
kernel32.dll 5.00.2195.2778 714.77 KB
(731,920 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\kernel32.dll
msvcrt.dll 6.10.8924.0 284.05 KB
(290,869 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\msvcrt.dll
winlogon.exe 5.00.2195.2953 173.77 KB
(177,936 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\winlogon.exe
sfcfiles.dll 5.00.2195.2967 948.27 KB
(971,024 bytes) 5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\sfcfiles.dll
ntdll.dll 5.00.2195.2779 478.77 KB (490,256 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\ntdll.dll
smss.exe 5.00.2195.2901 44.27 KB (45,328 bytes)
5/4/2001 12:05:02 PM
Microsoft Corporation
c:\winnt\system32\smss.exe

```

[Services]

Display Name	Name	State	Start Mode	Path	Error Control
Alerter	Alerter	Running	Auto	Share Process	
Application Management	AppMgmt	Stopped		Manual	LocalSystem
Computer Browser	Browser	Running	Auto	Share	Share
Indexing Service	ciSvc	Stopped	Manual	Share	Share
ClipBook	Clipsrv	Stopped	Manual	Own Process	Normal
Distributed File System	Dfs	Running	Auto		
DHCP Client	Dhcp	Running	Auto	Share	Share
Logical Disk Manager	Administrative Service	Stopped	Manual	Share Process	
Logical Disk Manager	dmserver	Running	Auto		
DNS Client	Dnscache	Running	Auto	Share	Share
Event Log	Eventlog	Running	Auto	Share Process	

COM+ Event System	EventSystem	Running			
Fax Service	Fax	Stopped	Manual	Own	Own
IIS Admin Service	IISADMIN	Running	Auto	Share	Share
InterSite Messaging	IsmSrv	Stopped	Disabled	Own	Own
Kerberos Key Distribution Center	kdc	Stopped	Disabled	Share Process	
Server Process	lanmanserver	Running	Auto	Share	Share
Workstation	lanmanworkstation	Running	Auto		
License Logging Service	LicenseService	Running	Auto	Own Process	
TCP/IP NetBIOS Helper Service	LmHosts	Running	Auto		
Messenger	Messenger	Running	Auto	Share Process	
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped	Manual	Own Process	
Distributed Transaction Coordinator	MSDTC	Running	Auto	Own Process	
Windows Installer	MSIserver	Stopped	Manual	Share	Share
Message Queuing	MSMQ	Running	Auto	Own	Own
Network DDE	NetDDE	Stopped	Manual	Share	Share
Network DDE DSDM	NetDDEdsdm	Stopped	Manual	Share Process	
Net Logon	Netlogon	Stopped	Manual	Share Process	
Network Connections	Netman	Running	Manual	Share	Share
Network News Transport Protocol (NNTP)	NntpSvc	Stopped	Auto	Share Process	
File Replication	NtFrs	Stopped	Manual	Own	Own
NT LM Security Support Provider	NtLmSsp	Running	Manual	Share Process	

```

c:\winnt\system32\lsass.exe Normal
LocalSystem 0
Removable Storage NtmsSvc Running Auto 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 Running Auto
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
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 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 Running Auto 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 Manual Share Process
c:\winnt\system32\services.exe
Normal LocalSystem 0
Distributed Link Tracking Client TrkWks
Running Auto 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 Auto Share Process
c:\winnt\system32\inetrv\inetinfo.exe
Normal LocalSystem 0
Windows Management Instrumentation WinMgmt
Running Auto Own Process
c:\winnt\system32\wbem\winmgmt.exe
Ignore LocalSystem 0
Windows Management Instrumentation Driver Extensions
Wmi Running Manual Share Process
c:\winnt\system32\services.exe
Normal LocalSystem 0

[Program Groups]

Group Name Name User Name
Accessories Default User:Accessories
Accessories\Accessibility Default Default User
Accessories\Entertainment Default Default User
Accessories\System Tools Default Default User
Accessories\System Tools\Startup Default Default User
Accessories All Users:Accessories All Users
Accessories\Accessibility All All Users
Users:Accessories\Accessibility All All Users
Accessories\Communications All All Users
Users:Accessories\Communications All All Users
Accessories\Entertainment All All Users
Users:Accessories\Entertainment All All Users
Accessories\Microsoft Script Debugger All All Users
Users:Accessories\Microsoft Script Debugger All All Users
Accessories\System Tools All All Users
Users:Accessories\System Tools All All Users
Administrative Tools All Users:Administrative Tools All Users
Startup All Users:Startup All Users

```

```

Accessories CLI01\Administrator:Accessories
CLI01\Administrator
Accessories\Accessibility
CLI01\Administrator:Accessories\Accessibility
CLI01\Administrator
Accessories\Entertainment
CLI01\Administrator:Accessories\Entertainment
CLI01\Administrator
Accessories\System Tools
CLI01\Administrator:Accessories\System Tools
CLI01\Administrator
Administrative Tools
CLI01\Administrator:Administrative Tools
CLI01\Administrator
Startup
CLI01\Administrator:Startup
CLI01\Administrator

[Startup Programs]

Program Command User Name Location
internat.exe internat.exe
CLI01\Administrator HKU\S-1-5-21-746137067-
1563985344-839522115-
500\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
internat.exe internat.exe .DEFAULT
HKU\DEFAULT\SOFTWARE\Microsoft\Windows\Curre
ntVersion\Run

[OLE Registration]

Object Local Server
Sound (OLE2) sndrec32.exe
Media Clip mplay32.exe
Video Clip mplay32.exe /avi
MIDI Sequence mplay32.exe /mid
Sound Not Available
Media Clip Not Available
Image Document "C:\Program Files\Windows
NT\Accessories\ImageVue\KodakImg.exe"
WordPad Document "%ProgramFiles%\Windows
NT\Accessories\WORDPAD.EXE"
Windows Media Services DRM Storage object Not
Available
Bitmap Image mspaint.exe

[Internet Explorer 5]

[ Following are sub-categories of this main category ]

[Summary]

Item Value
Version 5.00.3315.1000
Build 53315.1000
Product ID 51876-335-1829717-05787
Application Path C:\Program Files\Internet Explorer
Language English (United States)
Active Printer Not Available

Cipher Strength 168-bit
Content Advisor Disabled
IEAK Install No

[File Versions]

File Version Size Date Path
Company
advapi32.dll 5.0.2195.2867 352 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation

```

```

advpack.dll 5.0.3103.1000 87 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
browselc.dll 5.0.3315.2846 35 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
browseui.dll 5.0.3315.2846 789 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
ckcncv.exe 5.0.2189.1 9 KB
5:00:00 PM C:\WINNT\system32 Microsoft Corporation
comct132.dll 5.81.3103.1000 538 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
crypt32.dll 5.131.2195.2833 451 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
enhsig.dll <File Missing> Not Available Not Available
Available Not Available Not Available
iemigrat.dll <File Missing> Not Available Not Available
Available Not Available Not Available
iesetup.dll 5.0.3103.1000 57 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
iexplore.exe 5.0.2920.0 59 KB
12/7/1999 5:00:00 PM C:\Program Files\Internet Explorer Microsoft Corporation
imagehlp.dll 5.0.2195.2778 126 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
imghelp.dll <File Missing> Not Available Not Available
Available Not Available Not Available
inseng.dll 5.0.3103.1000 72 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
jobexec.dll 5.0.0.1 47 KB
5:00:00 PM C:\WINNT\system32 Microsoft Corporation
jscript.dll 5.1.0.5907 476 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
jsproxy.dll 5.0.2920.0 13 KB
12/7/1999 5:00:00 PM
C:\WINNT\system32 Microsoft Corporation
msahtml.dll <File Missing> Not Available Not Available
Available Not Available Not Available
mshtml.dll 5.0.3315.2870 2290 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
msjava.dll 5.0.3802.0 923 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
msoss.dll <File Missing> Not Available Not Available
Available Not Available Not Available
msxml.dll 8.0.5718.1 493 KB
12:05:02 PM C:\WINNT\system32 Microsoft Corporation
occache.dll 5.0.3103.1000 86 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
ole32.dll 5.0.2195.2887 970 KB
12:05:02 PM C:\WINNT\system32 Microsoft Corporation
oleaut32.dll 2.40.4517.0 612 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation

```

```

olepro32.dll 5.0.4517.0 160 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
rsabase.dll 5.0.2195.2228 128 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
rsaenh.dll 5.0.2195.2228 131 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
rsapi32.dll <File Missing> Not Available Not Available
Available Not Available Not Available
rsasig.dll <File Missing> Not Available Not Available
Available Not Available Not Available
schannel.dll 5.1.2195.0 138 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
shdoc401.dll <File Missing> Not Available Not Available
Available Not Available Not Available
shdocvw.dll 5.0.3315.2879 1078 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
shell32.dll 5.0.3315.2902 2304 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
shlwapi.dll 5.0.3315.1000 283 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
url.dll 5.0.2920.0 82 KB
5:00:00 PM C:\WINNT\system32 Microsoft Corporation
urlmon.dll 5.0.3315.1000 441 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
vbscript.dll 5.1.0.5907 428 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
webcheck.dll 5.0.3315.1000 252 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
win.com 5.0.2134.1 24 KB
5:00:00 PM C:\WINNT\system32 Microsoft Corporation
wininet.dll 5.0.3315.1000 457 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
winsock.dll 3.10.0.103 3 KB
12/7/1999 5:00:00 PM
C:\WINNT\system32 Microsoft Corporation
wintrust.dll 5.131.2195.2779 162 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
wsock.vxd <File Missing> Not Available Not Available
Available Not Available Not Available
wsock32.dll 5.0.2195.2871 21 KB
5/4/2001 12:05:02 PM
C:\WINNT\system32 Microsoft Corporation
wsock32n.dll <File Missing> Not Available Not Available
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 17492 MB
Available Disk Space 14838 MB
Maximum Cache Size 546 MB
Available Cache Size 547 MB

[List of Objects]
Program File Status CodeBase
No cached object information available

[Content]
[ Following are sub-categories of this main category ]
[Summary]
Item Value
Content Advisor Disabled

[Personal Certificates]
Issued To Issued By Validity Signature Algorithm
Administrator Administrator 1/31/2005 to 1/7/2105 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

```

Benchcraft Profile

Profile: 2500w

File Path: C:\Documents and Settings\Administrator\Desktop\2500w.pro
 Version: 4

Number of Engines: 4

Name: DRIVER1
 Description:
 Directory: c:\logfiles\rte02-1.log
 Machine: rte02
 Parameter Set: optim
 Index: 0
 Seed: 87340
 Configured Users: 6250
 Pipe Name: DRIVER13084375
 Connect Rate: 50
 Start Rate: 200
 Max. Concurrency: -1
 Concurrency Rate: 10
 CLIENT_NURAND: 233
 CPU: 0
 Additional Options:

Name: DRIVER2
 Description:
 Directory: c:\logfiles\rte02-2.log
 Machine: rte02
 Parameter Set: optim
 Index: 100000000
 Seed: 87340
 Configured Users: 6250
 Pipe Name: DRIVER23106406
 Connect Rate: 50
 Start Rate: 200
 Max. Concurrency: -1
 Concurrency Rate: 10
 CLIENT_NURAND: 233
 CPU: 1
 Additional Options:

Name: DRIVER3
 Description:
 Directory: c:\logfiles\rte03-1.log
 Machine: rte03
 Parameter Set: optim
 Index: 200000000
 Seed: 87340
 Configured Users: 6250
 Pipe Name: DRIVER33118796
 Connect Rate: 50
 Start Rate: 200
 Max. Concurrency: -1
 Concurrency Rate: 10
 CLIENT_NURAND: 233
 CPU: 0
 Additional Options:

Name: DRIVER4
 Description:
 Directory: c:\logfiles\rte03-2.log
 Machine: rte03
 Parameter Set: optim
 Index: 300000000
 Seed: 87340
 Configured Users: 6250
 Pipe Name: DRIVER43127671
 Connect Rate: 50
 Start Rate: 200
 Max. Concurrency: -1
 Concurrency Rate: 10
 CLIENT_NURAND: 233
 CPU: 1

Additional Options:

Number of User groups: 4

Driver Engine: DRIVER1
 IIS Server: cli01
 SQL Server: sql3040
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 1 - 625
 w_id Min Warehouse: 1
 w_id Max Warehouse: 2500
 Scale: Normal
 User Count: 6250
 District id: 1
 Scale Down: No

Driver Engine: DRIVER2
 IIS Server: cli01
 SQL Server: sql3040
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 626 - 1250
 w_id Min Warehouse: 1
 w_id Max Warehouse: 2500
 Scale: Normal
 User Count: 6250
 District id: 1
 Scale Down: No

Driver Engine: DRIVER3
 IIS Server: cli04
 SQL Server: sql3040
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 1251 - 1875
 w_id Min Warehouse: 1
 w_id Max Warehouse: 2500
 Scale: Normal
 User Count: 6250
 District id: 1
 Scale Down: No

Driver Engine: DRIVER4
 IIS Server: cli04
 SQL Server: sql3040
 Database: tpcc
 User: sa
 Protocol: HTML
 w_id Range: 1876 - 2500
 w_id Min Warehouse: 1
 w_id Max Warehouse: 2500
 Scale: Normal
 User Count: 6250
 District id: 1
 Scale Down: No

Number of Parameter Sets: 2

~Default
 Default Parameter Set

RT	RT	Menu	Txn	Think	Key
Time	Delay	Fence	Delay	Weight	Time
		New Order			
12.05	18.01	0.10	5.00	10.00	0.10

12.05	3.01	Payment	5.00	10.00	
		0.10			0.10
5.05	2.01	delivery	5.00	1.00	0.10
		0.10			
5.05	2.01	Stock Level	20.00	1.00	0.10
		0.10			
10.05	2.01	Order Status	5.00	1.00	0.10
		0.10			

optim

RT	RT	Menu	Txn	Think	Key
Time	Delay	Fence	Delay	Weight	Time
		New Order			
12.05	18.01	0.10	5.00	44.92	0.10
				43.03	
12.05	3.01	Payment	5.00	4.02	0.10
		0.10			
5.05	2.01	delivery	5.00	4.02	0.10
		0.10			
5.05	2.01	Stock Level	20.00	4.02	0.10
		0.10			
10.05	2.01	Order Status	5.00	4.02	0.10
		0.10			

Appendix D – 60-Day Space

TPC-C 60 Day Space Requirements						
Warehouses	2600			TpmC	31.253,00	
Table	Rows	Data KB	Index KB	Extra 5% KB	8hr Space	Total Space KB
Warehouse	2600	288	24	16		328
District	26000	2936	24	148		3108
Customer	78000000	56727280	3540000	3,013,364		63280644
History	78000000	4656728	16		911,472	4656744
NewOrder	23400000	416928	1072	20,900		438900
Orders	78000000	2546944	1242480		3,429,030	3789424
OrderLine	779997753	51999856	123080		11,000,052	52122936
Item	100000	9528	40	478		10046
Stock	260000000	83200000	176104	4,168,805		87544909
Total		199,560,488	5,082,840	7,203,711	15,340,554	211,847,039
		MB				
Dynamic Space	57,816	Sum of Data for Order, Orderline and History				
Static Space	149,066	Sum of Data+Index+5%-Dynamic Space				
Free Space	na	Total Allocated Spac - (Dynamic + Static Space)				
Daily Growth	11,120	(Dynamic Space/(W*62.5))*tpmc				
Daily Spread	-	(Free Space -1.5*Daily Growth) Zero Assumed				
60 Day Space MB	816,237					
60 Day Space GB	797,11					
Log Size	135,000,00					
KB Per New Order	5,17					
8 hr log MB	75,720					
8 hr log GB	73,9457					

	Misc fg	CS fg
	328	
	3108	
	0	63280644
	5568216	
	438900	
	7218454	
	63122988	
	10046	
	0	87544909
	76,362,040	150,825,553
files=	2	2
size=	6,732,800	12,800,000
Total=	13,465,600	25,600,000
* 8K blocks =	107,724,800	204,800,000
Needed =	76,362,040	150,825,553
OK		OK
		(53,974,447)

		Disks	Disks	Formatted Size	Space
Space Usage	GB Needed	Measured	Size	Size	Available
180 Day Space DB	797,11		18GB	16,90	-
		56,00	36GB	34,16	1,912,96
			4GB	4,00	-
Total DB		56,00			1,912,96
8-hr log + mirror	147,89	8,00	146GB	34,16	273,28
OS, Swap	3,00		36,00	34,16	-
Total Storage	948,00	GB			2,186,24

tpmC		31.253,00											
	Data Before KB	Index Before KB	Data After KB	Index After KB	Data Grow KB	Index Grow KB	Total Grow KB	KB/New-Order	8-Hr Growth KB	8-Hr Growth MB			
History	4,656,728	16	5134824	64	478,096	48	478,144	0,0608	911,471,95	890,11			
Order	2,546,944	1,242,480	3144016	2444224	597,072	1,201,744	1,798,816	0,2286	3,429,030,42	3,348,66			
Order-Line	51,999,856	123,080	57651984	241408	5,652,128	118,328	5,770,456	0,7333	11,000,051,78	10,742,24			
										14,981,01			
	sum(*) Before		sum(*) After		Num New-Order								
d_next_o_id	78,113,973		85,983,495		7,869,522								
	Before MB		After MB		Grow MB			KB/New-Order	8-Hr Growth MB	8-Hr Growth GB			
Log	1630,46		41352,20		39721,74			5,1687	75,720,40	73,95			
Database tpcc log used (%)								5,292,7315	bytes				
	135000	1,2077485		30,631262									