

**TPC Benchmark™ C
Full Disclosure Report
for
Dell PowerEdge 2650
Using
Microsoft SQL Server 2000 Standard Edition
and
Microsoft Windows 2003 Server**

Third Edition
Submitted for Review
Updated to meet TPC-C Version 5.3 specification and updated pricing

October 26, 2004

Fourth Printing, October 26, 2004

Dell 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, Dell is not responsible for any errors contained within this document.

The pricing information given in this FDR is accurate as of the publication date, October 26, 2004 and is generally available.

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. Actual performance experienced by a particular customer may vary due to differences in system layout and configuration, hardware and/or software revision levels, and background system activity. The content of this document is for informational purposes only.

Copyright 2004 Dell

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

PowerEdge is a trademark of Dell.

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

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

Intel and Pentium are registered trademarks of Intel Corporation.

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

Abstract

Overview

This report documents the methodology and results of the TPC Benchmark™ C test conducted on Dell PowerEdge 2650. The tests were run in a client/server configuration using one PowerEdge 1600SC as a client. The operating system used for the benchmark was Microsoft Windows 2003 Server on the database server and Microsoft Windows 2003 Server on the client. The database was Microsoft SQL Server 2000 Standard Edition. Microsoft COM+ provided the database connection queues. All tests were done in compliance with Revision 5.3 of the Transaction Processing Council's TPC Benchmark™ C Standard Specification. Two standard TPC Benchmark™ C metrics, transactions per second (tpmC) and price per tpmC (\$/tpmC) are reported and referred to in this document. The results from the tests are summarized below.

Hardware	Software	Total System Cost	tpmC	\$/tpmC	Availability Date
Dell PowerEdge 2650	Microsoft Windows 2003 Server with SQL Server 2000 Standard Edition	\$30,438	22,052	\$1.50	Feb 18, 2004

Auditor

The results of the benchmark and test methodology used to produce the results were audited by Lorna Livingtree of Performance Metrics and have fully met the TPC-C rev 5.3 specifications.

Additional copies of this Full Disclosure Report can be obtained from either the Transaction Processing Performance Council or Dell at the following address:

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

or

Dell
1 Dell Drive
Round Rock, TX 78682
Attention: Mike Molloy, Ph.D.



PowerEdge 2650

Client/Server w/1 PE1600SC Front End

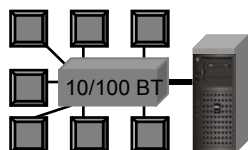
TPC-C Rev 5.3
Report Date
February 18, 2004
Revised Date
October 26, 2004

Total System Cost	TPC-C Throughput	Price/Performance	Availability Date	
\$30,438	22,052 tpmC	\$1.50 / tpmC	FEB 18, 2004	
Processors	Database Manager	OS	Other Software	Number of Users
1 x Intel Xeon™ Processors 3.2 GHz 2MB L3 Cache	Microsoft SQL Server 2000 Standard Edition	Microsoft Windows 2003 Server	Windows 2003 Server w/ COM+ Internet Information Server 5.0 Microsoft Visual C++	18,000

PE2650

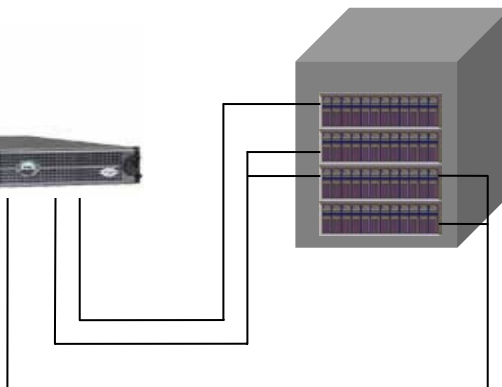
w/ 1 3.2 GHz Intel Xeon CPUs
w/ 2MB L3 cache,
2.5GB RAM,
2 PERC4-DC Controllers, and
1 On-board Adaptec AIC-7899 RAID
Controller
4 36GB 10K RPM U320 SCSI disks
2 Gigabit Ethernet adapters

4 PV220S Disk Pods
56 18GB 15K RPM U320 SCSI Disks



1 PowerEdge 1600SC Client

2 Intel Xeon @ 2.4GHz
w/ 512 KB L2
1024 MB RAM
1 36 GB Disk
2 Intel Pro100+ Ethernet NICs



System Component	Server		Each Client	
Processors Cache	1	Intel Xeon @ 3.2GHz 2MB cache	2	Intel Xeon w/ 512 KB L2 Client @ 2.4 GHz
Memory		2560 MB		1024 MB
Disk Controllers	2	PERC4-DC	1	Adaptec On-Board
	1	Adaptec AIC-7899 On-Board		
Disk Drives	56	18 GB SCSI	1	36 GB
	4	36 GB SCSI		
Total Storage		1100 GB		36 GB
Other	2	2GB NIC	2	10/100MB BT NIC
	1	CD-ROM		
	1	DAT		

Dell		PowerEdge 2650			TPC-C REV 5.3 EXECUTIVE SUMMARY		
					PAGE 2 OF 2		
		Client/Server			Report Date: 26-October-04		
Description	Part Number	Third Party	Unit Price	Qty	Extended Price	3 yr. Maint. Price	
Server Hardware		Brand	Pricing				
Dell PowerEdge 2650 3.2GHz/2MB	221-4490			1,531	1	1,531	290
PERC3/Di 128MB (2 int. ch)	340-3943			299	1	299	
2GB SDR AM, 4X512MB DIMMs	311-2733			1,099	1	1,099	
512MB RAM, 2 x 128MB DIMMS	311-3060			448	1	448	
Dell E773,17 in Gray (16.0 MS)	320-2907			184	1	184	
PERC4-DC 2-ch SCSI	340-8157			879	2	1,758	
					Subtotal	5,319	290
PowerVault Disk Subsystem							
PV220S, U3, PS, Tower	220-4477, etc.			1,471	4	5,884	1,604
2 SCSI Cables	310-0679			99	2	198	
18GB U320M SCSI 15K RPM Hard Drive	340-9472			249	56	13,944	
36GB U320M SCSI 10K RPM Hard Drive (OS+LOG)	340-9370			249	4	996	
					Subtotal	21,022	1,604
Server Software							
SQL Server 2000 St. Edition, Per processor licensing **	228-01079	Microsoft	1	4,999	1	4,999	
Windows 2003 Standard Server **	P73-00295	Microsoft	1	738	1	738	
Professional Support; per incident phone support		Microsoft	1	245	1		245
					Subtotal	5,737	245
Client Hardware							
Dell PowerEdge 1600SC, 2.4 GHz / 512KB L2/400 FSB	221-2207			227	1	227	290
Additional processor , 2.4 GHz / 512KB	311-2456			599	1	599	
1025MB RAM, 2 DIMMs	311-1940			548	1	548	
36GB U160M SCSI 10K RPM Hard Drive	340-7087			249	1	249	
Non-Redundant Power	310-1866			199	1	199	
IntelPro 100S	430-0369			59	1	59	
Dell E773,17 in Gray (16.0 MS)	320-2907			184	1	184	
					Subtotal	2,065	290
Client Software							
Windows 2003 Standard Server **	P73-00295	Microsoft	1	738	1	738	
Visual C++ ** .NET	254-00170	Microsoft	1	109	1	109	
					Subtotal	847	
User Connectivity							
7ft Crossover cable	CBLC5C7	LanAdapter	2	2	3	6	
					Subtotal	6	
					Other Discounts	(\$4,558)	
					Total	\$30,438	\$2,429
Notes: * Maint. included in PowerVault 220S disk pod or PV650F/630F fibre channel disk pod					Three-Year Cost of Ownership: \$32,867		
** All Microsoft maintenance is covered by the maintenance costs of Microsoft SQL Server					tpmC Rating: 22052		
*** 10% or minimum 2 spares are added in place of onsite service (products have a five year return-to-vendor warranty) Pricing: 1 - Microsoft 2 - LanAdapter					\$ /tpmC: 1.50		
Pricing may be verified by calling 1-800-BUY-DELL and referencing quote # 180780429 as a complex quote.							
Audited by Loma Livingtree, 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 .							

MQTh, computed Maximum Qualified Throughput

22,052 tpmC

Response Times (in seconds)

	Average	90 th	Max
- Neworder	0.99	1.41	6.60
- Payment	0.62	0.99	4.12
- Delivery (interactive portion)	0.12	0.17	0.64
- Stock-Level	1.78	4.67	11.55
- Order Status	0.85	1.25	6.08
- Delivery (deferred portion)	1.81	2.50	4.39
- Menu	0.12	0.17	0.80

Response time delay added for emulated components

Menu 0.1
Resp 0.1

Transaction Mix, in percent of total transactions

- New-Order	44.84%
- Payment	43.02 %
- Delivery	4.04 %
- Stock-Level	4.05 %
- Order-Status	4.05 %

Keying/Think Times (in seconds),

	Min		Average		Max	
- New-Order	18.02	0.0	18.02	12.03	18.05	120.42
- Payment	3.01	0.0	3.02	12.03	3.05	120.41
- Delivery	2.01	0.0	2.02	5.04	2.04	50.40
- Stock-Level	2.01	0.0	2.02	5.05	2.05	50.40
- Order-Status	2.01	0.0	2.02	10.04	2.04	100.41

Test Duration

- Ramp-up time	5 minutes
- Measurement interval	120 minutes
- Number of checkpoints	4
- Checkpoint interval	30 minutes
- Number of transactions (all types) completed in measurement interval	6,140,323

Table of Contents

ABSTRACT	1
OVERVIEW	1
AUDITOR.....	1
TABLE OF CONTENTS	1
INTRODUCTION	6
DOCUMENT STRUCTURE	6
BENCHMARK OVERVIEW	6
SYSTEM OVERVIEW	7
GENERAL ITEMS	8
TEST SPONSOR.....	8
APPLICATION CODE AND DEFINITION STATEMENTS	8
PARAMETER SETTINGS	8
CONFIGURATION DIAGRAMS.....	9
CLAUSE 1 -- LOGICAL DATABASE DESIGN RELATED ITEMS	11
TABLE DEFINITIONS	11
PHYSICAL ORGANIZATION OF THE DATABASE	11
INSERT AND DELETE OPERATIONS.....	11
HORIZONTAL AND VERTICAL PARTITIONING.....	11
REPLICATION	11
TABLE ATTRIBUTES	11
CLAUSE 2 -- TRANSACTION AND TERMINAL PROFILES RELATED ITEMS	12
RANDOM NUMBER GENERATION	12
SCREEN LAYOUT	12
TERMINAL VERIFICATION.....	12
INTELLIGENT TERMINALS.....	12
TRANSACTION PROFILES	12
TRANSACTION MIX	13
DEFERRED DELIVERY MECHANISM	13
CLAUSE 3 -- TRANSACTION AND SYSTEM PROPERTIES RELATED ITEMS	14
ACID TESTS	14
<i>Atomicity</i>	14
<i>Consistency</i>	14
<i>Isolation</i>	14
<i>Durability</i>	15
CLAUSE 4 -- SCALING AND DATABASE POPULATION RELATED ITEMS	17
TABLE CARDINALITY	17
CONSTANT VALUES	17
DATA DISTRIBUTION	18
PARTITION MAPPING.....	20
60 DAY SPACE CALCULATION.....	20
CLAUSE 5 -- PERFORMANCE METRICS AND RESPONSE TIME RELATED ITEMS	22

MEASURED TPMC	22
RESPONSE TIMES.....	22
THINK TIMES & KEY TIMES.....	22
RESPONSE TIME DISTRIBUTION CURVES	22
NEW-ORDER THINK TIME DISTRIBUTION GRAPH	26
STEADY-STATE GRAPH	27
STEADY-STATE METHODOLOGY.....	27
WORK PERFORMED DURING STEADY STATE	27
MEASUREMENT INTERVAL	28
MEASUREMENT PERIOD DURATION AND CHECKPOINT DURATION.....	28
TRANSACTION MIX	28
OTHER METRICS	29
RTE PARAMETERS.....	30
EMULATED COMPONENTS.....	30
BENCHMARKED AND TARGETED SYSTEM CONFIGURATION DIAGRAMS.....	30
NETWORK CONFIGURATION	30
NETWORK BANDWIDTH	30
OPERATOR INTERVENTION.....	31
CLAUSE 7 -- PRICING RELATED ITEMS	32
HARDWARE AND SOFTWARE LIST	32
AVAILABILITY DATE.....	32
MEASURED TPMC	32
COUNTRY SPECIFIC PRICING	32
USAGE PRICING	32
SYSTEM PRICING.....	33
CLAUSE 9 -- AUDIT RELATED ITEMS	34
AUDITOR.....	34
AVAILABILITY OF THE FULL DISCLOSURE REPORT	34
APPENDIX A - APPLICATION SOURCE CODE	37
TPCC.DLL ISAPI DLL SOURCE CODE	37
<i>isapi_dll/src/tpcc.def</i>	37
<i>isapi_dll/src/tpcc.h</i>	37
<i>isapi_dll/src/tpcc.rc</i>	39
<i>isapi_dll/src/tpcc.cpp</i>	40
<i>isapi_dll/src/resource.h</i>	61
<i>common/src/ReadRegistry.cpp</i>	61
<i>common/src/ReadRegistry.h</i>	62
<i>common/src/error.h</i>	63
<i>common/src/trans.h</i>	65
<i>common/src/txn_base.h</i>	67
<i>db_dblib_dll/src/tpcc_dblib.cpp</i>	67
<i>db_dblib_dll/src/tpcc_dblib.h</i>	77
<i>tm_com_dll/src/tpcc_com.cpp</i>	78
<i>tm_com_dll/src/tpcc_com.h</i>	80
<i>tpcc_com_all/src/methods.h</i>	81
<i>tpcc_com_all/src/resource.h</i>	83
<i>tpcc_com_all/src/tpcc_com_all.cpp</i>	83
<i>tpcc_com_all/src/tpcc_com_all.def</i>	88
<i>tpcc_com_all/src/tpcc_com_all.h</i>	88
<i>tpcc_com_all/src/tpcc_com_all.idl</i>	89
<i>tpcc_com_all/src/tpcc_com_all.rc</i>	90

<i>tpcc_com_all/src/tpcc_com_all.rgs</i>	91
<i>tpcc_com_all/src/tpcc_com_all_i.c</i>	91
<i>tpcc_com_all/src/tpcc_com_no.rgs</i>	93
<i>tpcc_com_all/src/tpcc_com_os.rgs</i>	93
<i>tpcc_com_all/src/tpcc_com_pay.rgs</i>	94
<i>tpcc_com_all/src/tpcc_com_ps.h</i>	94
<i>tpcc_com_all/src/tpcc_com_sl.rgs</i>	96
<i>tpcc_com_ps/src/dlldata.c</i>	97
<i>tpcc_com_ps/src/tpcc_com_ps.def</i>	97
<i>tpcc_com_ps/src/tpcc_com_ps.h</i>	97
<i>tpcc_com_ps/src/tpcc_com_ps.idl</i>	100
<i>tpcc_com_ps/src/tpcc_com_ps_i.c</i>	100
<i>tpcc_com_ps/src/tpcc_com_ps_p.c</i>	101
<i>common/txnlog/include/rtetime.h</i>	122
<i>common/txnlog/include/spinlock.h</i>	122
<i>common/txnlog/include/txnlog.h</i>	123

APPENDIX B - DATABASE DESIGN.....127

BUILD SCRIPTS	127
<i>setup.cmd</i>	127
<i>tables.sql</i>	129
<i>idxcuscl.sql</i>	130
<i>idxcusnc.sql</i>	130
<i>idxdiscl.sql</i>	131
<i>idxitmcl.sql</i>	131
<i>idxnodcl.sql</i>	131
<i>idxodlcl.sql</i>	131
<i>idxordcl.sql</i>	132
<i>idxstkcl.sql</i>	132
<i>idxwarcl.sql</i>	132
<i>dbopt1.sql</i>	132
<i>dbopt2.sql</i>	133
<i>dbopt3.sql</i>	133
<i>backup.sql</i>	133
-- File: <i>BACKUP.SQL</i>	134
-- Microsoft TPC-C Benchmark Kit Ver. 4.22	134
-- Copyright Microsoft, 2001.....	134
-- Purpose: Creates backup of tpcc database.....	134
<i>declare @startdate datetime</i>	134
<i>declare @enddate datetime</i>	134
<i>select @startdate = getdate()</i>	134
<i>select "Start date:", convert(varchar(30),@startdate,9)</i>	134
<i>backup database tpcc to tpccback1, tpccback2 with init, stats = 1</i>	134
<i>select @enddate = getdate()</i>	134
<i>select "End date: ", convert(varchar(30),@enddate,9)</i>	134
<i>select "Elapsed time (in seconds): ", datediff(second, @startdate,</i> <i>@enddate)</i>	134
<i>go</i>	134
<i>restore.sql</i>	134
STORED PROCEDURES	135
<i>neword.sql</i>	135
<i>payment.sql</i>	137
<i>ordstat.sql</i>	139

<i>delivery.sql</i>	140
<i>stocklev.sql</i>	141
LOADER SOURCE CODE.....	141
<i>tpcc.h</i>	141
<i>tpccldr.c</i>	143
<i>getargs.c</i>	163
<i>random.c</i>	164
<i>strings.c</i>	166
<i>time.c</i>	169
APPENDIX C - TUNABLE PARAMETERS	170
SERVER CONFIGURATION PARAMETERS.....	170
<i>Microsoft Windows 2003 Server Parameters</i>	170
<i>Microsoft Windows 2003 Server Configuration</i>	170
<i>Microsoft SQL Server 2000 Startup Parameters</i>	170
<i>Microsoft SQL Server Stack Size</i>	171
<i>Microsoft SQL Server 2000 Configuration Parameters</i>	171
<i>TPCC Application Registry Parameters</i>	271
<i>Windows Registry Editor Version 5.00</i>	271
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\TPCC].....	271
"Path"="C:\Inetpub\wwwroot\".....	271
"NumberOfDeliveryThreads"=dword:0000004b.....	271
"MaxConnections"=dword:000061a8.....	271
"MaxPendingDeliveries"=dword:0000012c.....	271
"DB_Protocol"="ODBC".....	271
"TxnMonitor"="COM".....	271
"DbServer"="pe2650".....	271
"DbName"="tpcc".....	271
"DbUser"="sa".....	271
"DbPassword"="".....	271
"COM_SinglePool"="YES".....	271
<i>Microsoft Internet Information Server Registry Parameters</i>	271
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo].....	271
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters].....	271
"ListenBackLog"=dword:00000019.....	271
"DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,00,00.....	271
"PoolThreadLimit"=dword:000000be.....	271
"ThreadTimeout"=dword:00015180.....	271
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Performance].....	271
"Library"="infectrs.dll".....	271
"Open"="OpenINFOPerformanceData".....	271
"Close"="CloseINFOPerformanceData".....	271
"Collect"="CollectINFOPerformanceData".....	271
"Last Counter"=dword:00000842.....	271
"Last Help"=dword:00000843.....	271
"First Counter"=dword:00000802.....	271
"First Help"=dword:00000803.....	271
"Library Validation Code"=hex:de,fc,ed,18,0a,98,c0,01,10,25,00,00,00,00,00,00.....	271

"WbemAdapFileTime"=hex:00,60,4e,96,aa,40,bf,01.....	271
"WbemAdapFileSize"=dword:00002510.....	271
"WbemAdapStatus"=dword:00000000.....	271
<i>World Wide Web Service Registry Parameters</i>	271
RTE INPUT PARAMETERS	273
<i>BenchCraft Configuration File</i>	273
APPENDIX D – DISK STORAGE	280
APPENDIX E - PRICE QUOTATIONS	281

Introduction

Document Structure

The TPC Benchmark C Standard Specification Revision 5.3, written and approved by the Transaction Processing Performance Council (TPC), determines the contents of this report. The format of this report is based on this specification. Most sections of this report begin with the specification requirements printed in italic type, immediately followed by the detail in plain type of how Dell complied with the specification. Where extensive listings are required (such as listing of code), a note is included which references an appendix containing the listing.

Benchmark Overview

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.

System Overview

The hardware configuration used in this TPC-C test is a Dell PowerEdge 2650 server driven by one Dell PowerEdge 1600 client. The PE2650 has two internal Gigabit Ethernet adapters, of which only one is in use. The client and server are networked together via a cross-over cable. Five remote terminal emulator (RTE) systems (PowerEdge 2200's) emulate 16,200 users executing the standard TPC-C workload. The RTE's are connected to the client through a 10/100 BaseT switch. The switch connects to the client machine at 100 BaseT and to the RTE machines at 10Mbit/sec, half duplex. Microsoft Windows 2003 Server was the operating system used on the server. Microsoft Windows 2003 Server was used on the client. Microsoft SQL Server 2000 Standard Edition was the database on the server machine.

The PowerEdge 2650 motherboard uses the ServerWorks Grand Champion High End chipset and can hold up to two Pentium® 4 Xeon processors (3.2 GHz with 2 MB L2 cache each). The system has 3 PCI-X 64-bit/100MHz I/O slots and a single legacy 32-bit/33MHz PCI slot. The measured configuration used 2.50 Gbytes of DDR RAM, which was achieved by using four 512 Mbyte DIMMs and two 256Mbyte DIMMs.

The PowerEdge 2650 has an integrated Adaptec AIC-7899 U160 SCSI controller to which was attached 4 36GB hard disks in RAID 10 configuration containing the database log and OS. In addition, three DELL PERC3 2-channel RAID controllers were installed in PCI-X slots. The three PERC3 controllers were connected to four PV220 disk pods enclosing a total of 56 18GB 15K RPM SCSI disks, containing database data. There was one empty PCI-X slots.

The client has dual 2.4GHz Intel Xeon processors with 512 Kbytes of L2 cache. The client has 1024 Mbytes of RAM, one 18 GB hard disk, one intergrated Intel Ether Express Pro100+ PCI Ethernet adapter and one Intel Pro 100 Network Interface Card. The client's Intel Ethernet adapter was connected to the RTE machines through a 10/100 BaseT switch and the Intel Pro NIC was connected to the Database Server through a cross-over cable. The client was driven through twelve network segments to run a total of 18,000 emulated users. The network segments between the switches and RTEs were fixed at 10 Mbit/sec, half duplex.

General Items

Test Sponsor

A statement identifying the sponsor of the Benchmark and any other companies who have participated.

Dell was the test sponsor of this TPC Benchmark™ C.

Application Code and Definition Statements

The application program must be disclosed. This includes, but is not limited to, the code implementing the five transactions and the terminal input/output functions.

The application consists of the Microsoft Benchcraft Remote Terminal Emulator (RTE) program emulating a set of users entering TPC-C transactions through web browsers, and communicating with Client machines running the Microsoft Internet Information Server (IIS) web server. The Client machines use the COM+ transaction monitor to communicate with the database server machine.

On each Client machine IIS loads a custom Microsoft Internet Information Server Application Programming Interface dynamic link library (ISAPI DLL) application program that communicates with the emulated web browsers through the HTTP protocol and with the database server through the COM+ transaction monitor and the Microsoft DBLIB interface. The application supplies fill-in screens to the user for each transaction, then parses the data in each request, and makes a call on SQL Server through the COM+ layer, which manages a set of DBLIB connections to the database server. The resulting data is passed back to the application where it is formatted into HTML and sent back to the user's browser. The Delivery transaction is handled directly from the application to the database without the use of COM+.

The web Client code is listed in Appendix A.

Parameter Settings

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

- *Database options*
- *Recover/commit options*
- *Consistency/locking options*
- *System parameter, application parameters, and configuration parameters.*

This requirement can be satisfied by providing a full listing of all parameters and options.

Appendix C contains all the database, Windows 2003 Server, and Internet Information Service parameters used in this benchmark.

Appendix D contains the 60 day space calculations.

Configuration Diagrams

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

Figures 1 and 2 respectively show the measured and priced full client/server configurations. The system under test (SUT) in the measured system was identical to what was priced.

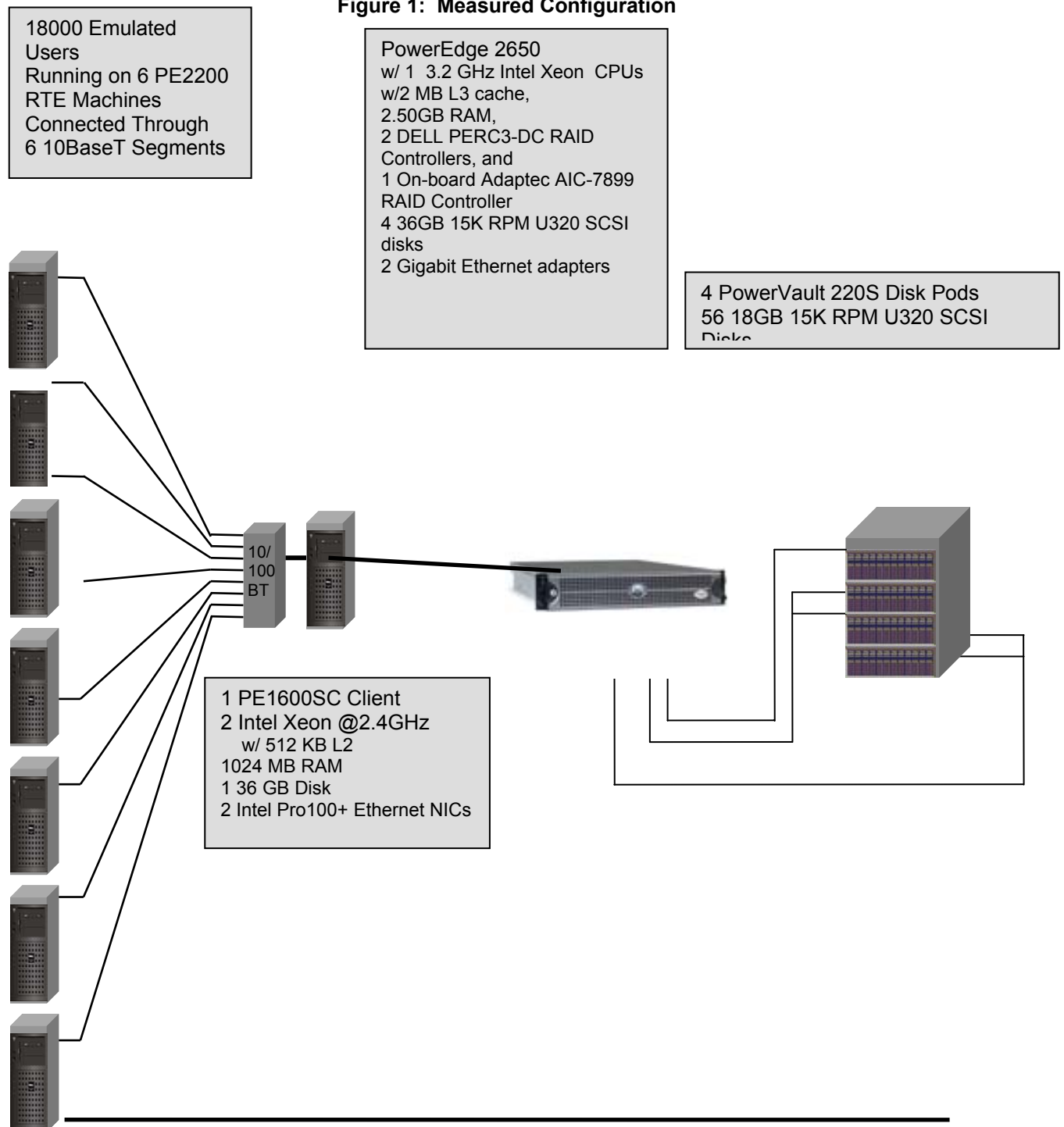
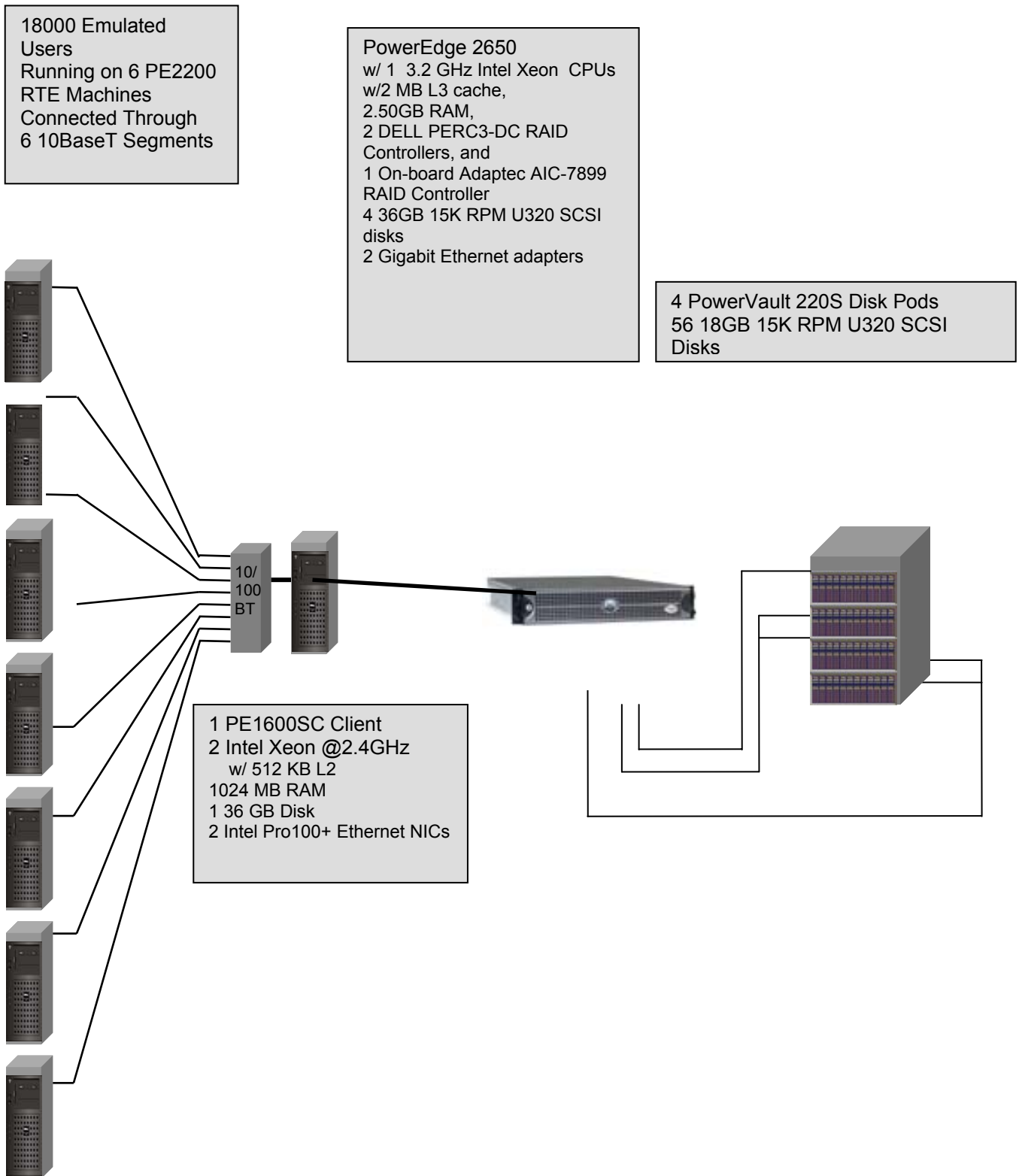


Figure 2: Priced Configuration



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. (8.1.2.1)

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. (8.1.2.2)

The measured configuration used 60 disk drives. The organization is shown in Table 5: 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. (8.1.2.3)

Insert and delete functionality was fully operational during the benchmark.

Horizontal and Vertical Partitioning

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

Partitioning was not used in this benchmark.

Replication

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

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). (8.1.2.6)

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 described. (8.1.3.1)

The random number generation was done internal to the Microsoft BenchCraft RTE program, which was audited independently.

Screen Layout

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

The screen layouts are based on those in Clauses 2.4.3, 2.5.3, 2.6.3, 2.7.3, and 2.8.3 of the TPC-C Standard Specification. There are some very minor differences based on the fact that this is a web client implementation.

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). (8.1.3.3)

The terminal features were verified by allowing the auditor to manually execute each of the five transaction types, using Microsoft Internet Explorer version 3.0.

Intelligent Terminals

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

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.*

Application code involved in the manipulation of data was run on the client. Screen manipulation commands in the form of HTML were downloaded to the web browser, which handled input and output presentation graphics. A listing of this code is included in Appendix A. Microsoft Internet Information Service assisted in the processing and presentation of this data.

Transaction Profiles

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

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

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

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

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

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

Table 1: Transaction Statistics

Transaction	Function	Value
New Order	Home Warehouse Items	99.00%
	Remote Warehouse Items	1.00%
	Rolled Back Transactions	0.99%
	Average Lines Per Order	10.00
Payment	Home Warehouse	84.99%
	Remote Warehouse	15.01%
	Non-Primary Key Access	60.02%
Order Status	Non-Primary Key Access	60.07%
Delivery	Skipped Transactions	0

Transaction Mix

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

Table 2: Transaction mix

Transaction	Percentage
New Order	44.84%
Payment	43.02%
Order Status	4.04%
Delivery	4.05%
Stock Level	4.05%

Deferred Delivery Mechanism

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

The application creates a semaphore-base thread pool consisting of a user-specified number of threads, which open DBLIB connections on the database. When a Delivery transaction is posted one of these threads makes the database call while the transaction's original thread returns control to the user. Upon completion the Delivery 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

ACID Tests

The results of the ACID tests must be disclosed along with a description of how the ACID requirements were met. This includes disclosing which case was followed for the execution of Isolation Test 7. (8.1.4.1)

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

Atomicity

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

Completed Transactions

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

Aborted Transactions

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

Consistency

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

Consistency conditions one through four were tested using a shell script to issue queries to the database. The results of the queries verified that the database was consistent for all four tests. A run was executed under full load lasting over ten (10) minutes and included a checkpoint. The shell script was executed again. The result of the same queries verified that the database remained consistent after the run.

Isolation

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

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

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

For Isolation test seven, case A was followed.

Durability

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

Durable Media Failure

Durability from media failure was demonstrated on the 1900 warehouse database. The standard driving mechanism was used to generate the transaction load of 18000 users for the Loss of Data.

Loss of Data

Loss of data was demonstrated on the 1900 Warehouse database. The standard driving mechanism was used to generate the transaction load of 18000 users for the test. To demonstrate recovery from a permanent failure of durable media containing TPC-C tables, the following steps were executed:

1. The 1900 Warehouse database was used for this test.
2. The database was backed up using SQL Server backup facilities.
3. A sum of D_NEXT_O_ID was taken.
4. 18000 users were logged in to the database and ran transactions.
5. One disk drive in the data array was removed causing SQL Server errors.
6. The RTE was allowed to continue running. Completed transactions enroute from the clients were recorded. Error messages began appearing on the RTE screen.
7. The RTE was stopped.
8. SQL Server was stopped and restarted and a dump of the transaction log was taken.
9. SQL Server was stopped, Windows 2003 was shutdown and the machine powered off.
10. The failed disk was replaced.
11. The machine was powered up, Windows 2003 and SQL Server were started.
12. The TPC-C database was dropped and restored from backup.
13. The transaction log was restored and transactions rolled forward.
14. A new count of D_NEXT_O_ID was taken.
15. This number was compared with the number of new orders reported by the RTE. The difference was valid per the spec.

Instantaneous Interruption and Loss of Memory/Loss of Log

Instantaneous Interruption and Loss of Memory were demonstrated on the full database with 1900 warehouses in a single test. The standard driving mechanism was used to generate the transaction load of 18000 users for the test. To demonstrate recovery an instantaneous system interruption caused by powering off the Server, the following steps were executed:

1. The full database was used.
2. A sum of D_NEXT_O_ID was taken.
3. 18000 users were logged in to the database and ran transactions.
4. The system was run in steady state for 5 minutes
5. One disk drive in the transaction log array was removed with no effect on Windows 2003 or SQL Server.
6. The system ran for an additional 5 minutes.
6. The Server was powered off by normal means, causing instantaneous interruption.

7. The RTE was allowed to continue running. Completed transactions enroute from the clients were recorded. Error messages began appearing on the RTE screen.
8. The RTE was stopped.
9. The server was powered on again and rebooted.
10. SQL Server was restarted and automatically recovered.
11. A new count of D_NEXT_O_ID was taken.
12. This number was compared with the number of new orders reported by the RTE. The difference was valid per the spec.

Clause 4 -- Scaling and Database Population Related Items

Table Cardinality

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

The database was originally built with 1900 warehouses. The performance run used 1800 warehouses and this was verified by runcheck.

Table 3: Table Cardinality

Table	Cardinality as Benchmarked
Warehouse	1,900
District	19,000
Customer	57,000,000
History	57,000,000
NewOrder	17,100,000
Orders	57,000,000
OrderLine	570,001,928
Item	100,000
Stock	190,000,000
Deleted Warehouses	0

Constant Values

The following values were used as constant value inputs to the NURand function for this benchmark.

Table 4: Constant Values

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

Data Distribution

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

The Database was built using a total of 60 disks: 56 18GB for data, 4 36GB for log and OS and application software. The data drives were configured as hardware RAID 0. Logs and OS were configured as hardware RAID 10. DELL PERC4-DC RAID Controllers 2 and 3 were configured with 1 logical drive each. Each logical drive spanned 28 disk drives. PERC3/Di intergrated controller 1 was configured with 1 logical drive spanning 4 36GB SCSI drives. Each Windows 2003 data drive contained 3 partitions: partition 1 for customer/stock, partition 2 for miscellaneous, and partition 3 for backup. Partitions 1 and 2 were RAW file systems and partition 3 was formatted NTFS. The details are shown in Table 5.

Table 5: Data Distribution

W2K Disk Administration	Adaptec 7899 Configuration				
Disk 2 8663MB	On-Board Controller # 1				
Partition	On-Board		Channels		
1	Internal	SCSI ID	0		
C: OS NTFS 8663 MB		0	A0-1		
		1			
		2			
		3			

W2K Disk Administration	Adaptec 7899 Configuration				
Disk 2 69417MB	On-Board Controller # 1				
Partition	On-Board		Channels		
1	Internal	SCSI ID		1	
L: LOG RAW 57354 MB		0		A1-1	
		1		A1-2	
		2		A1-3	
		3		A1-4	

W2K Disk Administration			DELL PERC4-DC Configuration					
Disk 0 483779MB			Controller # 2					
Partition			Slot# 1		Channels			
1	2	3		SCSI ID	A	B	C	D
K: CS1 RAW 156.26GB	Y: MS1 RAW 156.26GB	U: Backup1 B1 NTFS 159.93GB		0	A1-1	A2-1		
				1	A1-2	A2-2		
				2	A1-3	A2-3		
				3	A1-4	A2-4		
				4	A1-5	A2-5		
				5	A1-6	A2-6		
				8	A1-7	A2-7		
				9	A1-8	A2-8		
				10	A1-9	A2-9		
				11	A1-10	A2-10		
				12	A1-11	A2-11		
				13	A1-12	A2-12		
				14	A1-13	A2-13		
				15	A1-14	A2-14		

W2K Disk Administration			DELL PERC4-DC Configuration					
Disk 1 483779MB			Controller # 3					
Partition			Slot# 2		Channels			
1	2	3		SCSI ID	A	B	C	D
S: CS2 RAW 156.26GB	W: MS2 RAW 156.26GB	V: Backup2 B2 NTFS 159.93GB		0	A1-1	A2-1		
				1	A1-2	A2-2		
				2	A1-3	A2-3		
				3	A1-4	A2-4		
				4	A1-5	A2-5		
				5	A1-6	A2-6		
				8	A1-7	A2-7		
				9	A1-8	A2-8		
				10	A1-9	A2-9		
				11	A1-10	A2-10		
				12	A1-11	A2-11		
				13	A1-12	A2-12		
				14	A1-13	A2-13		
				15	A1-14	A2-14		

Comment: Detailed diagrams for layout of database files on disks can widely vary, and it is difficult to provide exact guideline suitable for all implementations. The intent is to provide sufficient detail to allow independent reconstruction of the test database. The two figures below are examples of database layout descriptions and are not intended to depict or imply any optimal layout for the TPC-C database.

8.1.5.3 A statement must be provided that describes:

1. The data model implemented by the DBMS used (e.g., relational, network, hierarchical)
2. The database interface (e.g., embedded, call level) and access language (e.g., SQL, DL/1, COBOL read/write) used to implement the TPC-C transactions. If more than one interface/access language is used to implement TPC-C, each interface/access language must be described and a list of which interface/access language is used with which transaction type must be disclosed.

Microsoft SQL Server 2000 Standard 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 DBLIB interface.

Partition Mapping

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

Comment: The intent is to provide sufficient detail about partitioning and replication to allow independent reconstruction of the test database. (8.1.5.4)

An description of a database partitioning scheme is presented below as an example. The nomenclature of this example was outlined using the CUSTOMER table (in Clause 8.1.2.1), and has been extended to use the ORDER and ORDER_LINE tables as well.

The database was not replicated.

60 day Space Calculation

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). (8.1.5.5)

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 usage was determined by running *dbcc sqlperf(logspace)*
2. Transactions were run against the database with a full load of users.
3. The final log space usage was determined by running *dbcc sqlperf(logspace)*
4. The space used was calculated as the difference between the first and second query.
5. The number of NEW-ORDERS was verified from an RTE report covering the entire run.
6. The space used was divided by the number of NEW-ORDERS giving a spaceused per NEW-ORDER transaction.
7. The space used per transaction was multiplied by the measured tpmC rate times 480 minutes.

The results of the above steps yielded a requirement of 100.2906 GB (including mirror) to sustain the log for 8 hours. Space available on the transaction log volume was 134.98GB (including mirror), indicating that enough storage was configured to sustain 8 hours of growth.

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

The details of the 60-day space requirement is shown in Appendix D.

Clause 5 -- Performance Metrics and Response Time Related Items

Measured TpmC

Measured tpmC must be reported. (8.1.6.1)

Measured TpmC 22,052
Price per TpmC \$1.50

Response Times

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

Table 6: Transaction Response Times

Transaction	Average	90%	Maximum
New Order	0.99	1.41	6.60
Payment	0.62	0.99	4.12
Interactive Delivery	0.12	0.17	0.64
Stock Level	1.78	4.67	11.55
Order Status	0.85	1.25	6.08
Deferred Delivery	1.81	2.50	4.39
Menu	0.12	0.17	0.80

Think Times & Key Times

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

Table 7: Transaction Key Times

Transaction	Minimum	Average	Maximum
New Order	18.02	18.02	18.05
Payment	3.01	3.02	3.05
Delivery	2.01	2.02	2.04
Stock Level	2.01	2.02	2.05
Order Status	2.01	2.02	2.04

Table 8: Transaction Think Times

Transaction	Minimum	Average	Maximum
New Order	0.00	12.04	120.42
Payment	0.00	12.03	120.41
Delivery	0.00	5.04	50.40
Stock Level	0.00	5.05	50.40
Order Status	0.00	10.04	100.41

Response Time Distribution Curves

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

transaction type. (8.1.6.4)

Figure 3: New Order Response Time Distribution

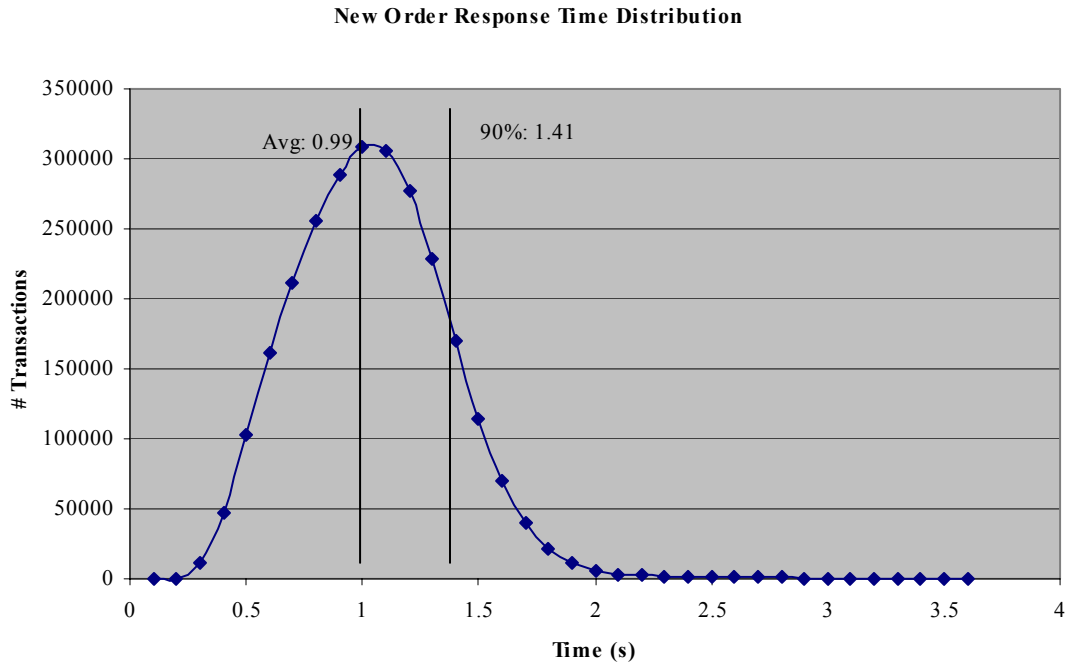


Figure 4: Payment Response Time Distribution

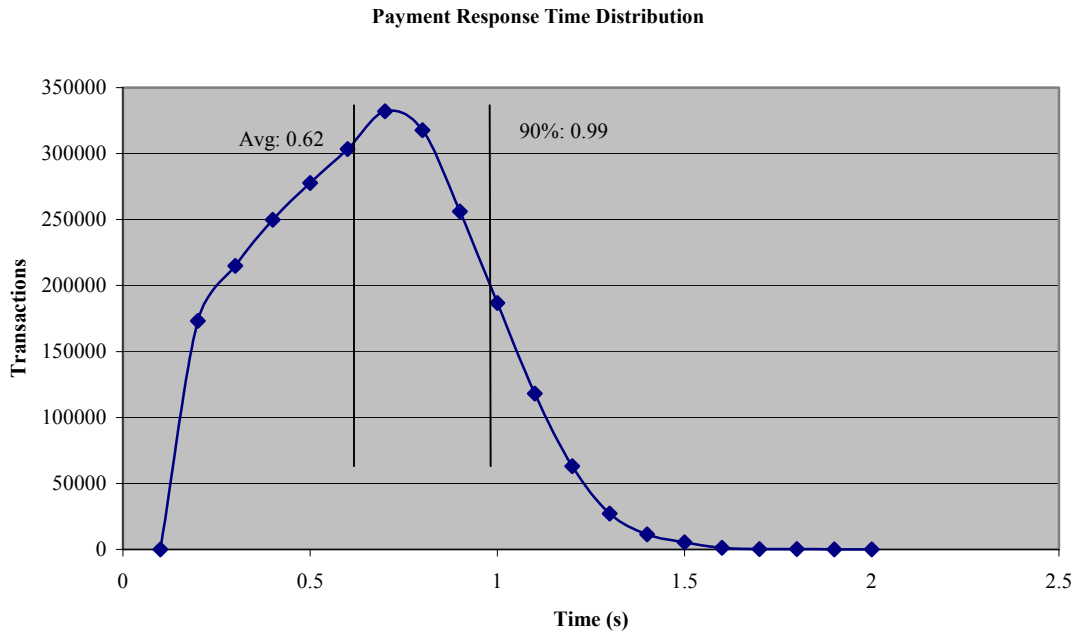


Figure 5: Order Status Response Time Distribution

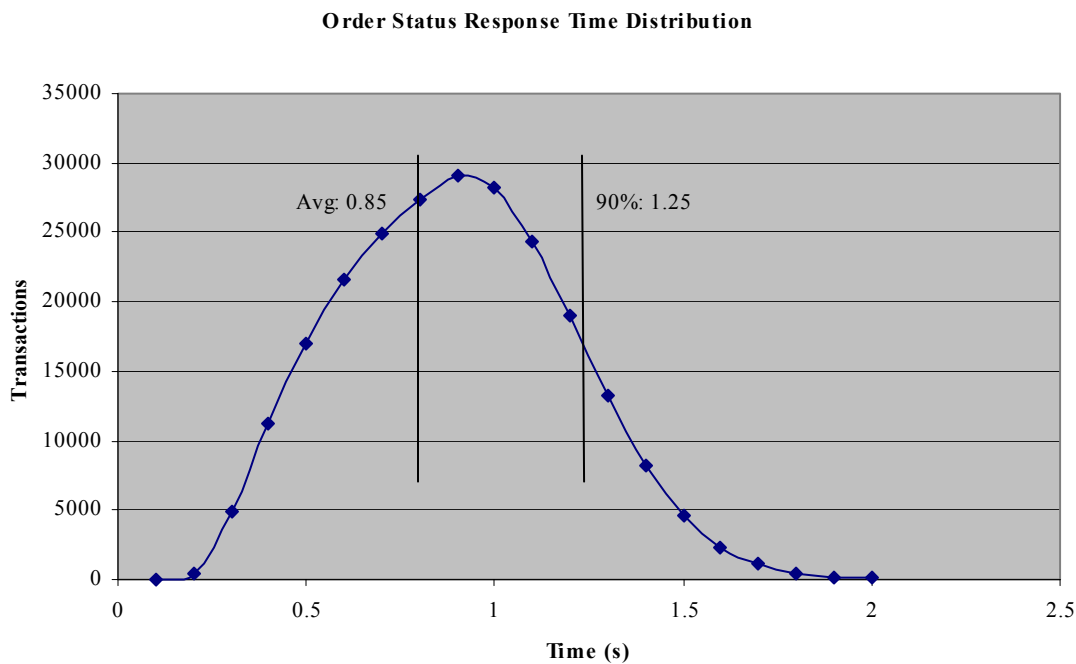


Figure 6: Delivery Response Time Distribution

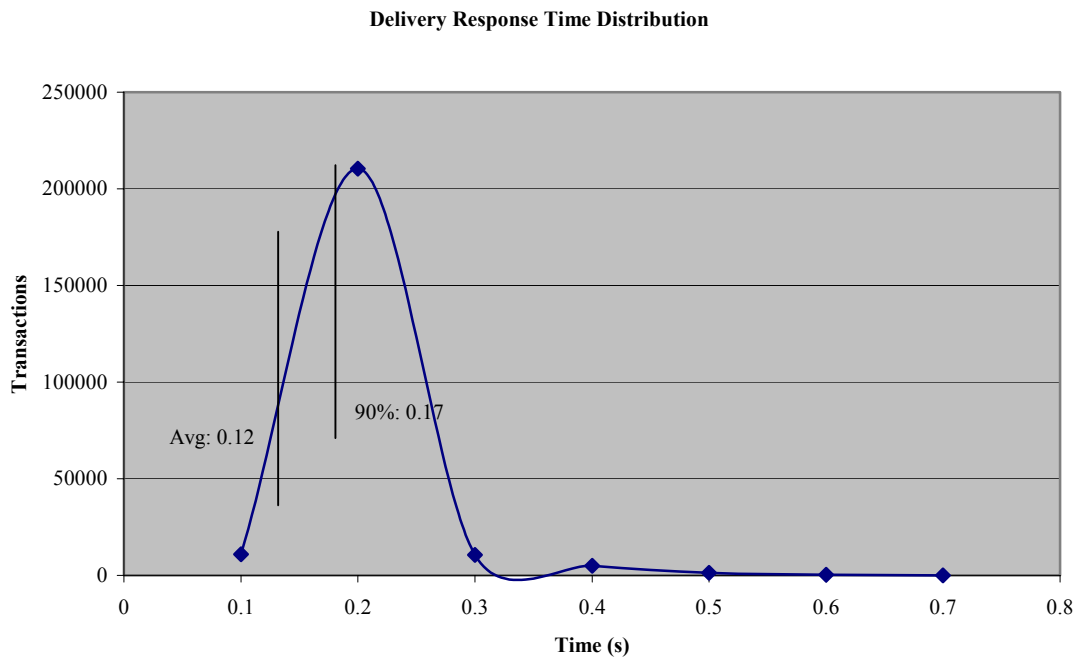
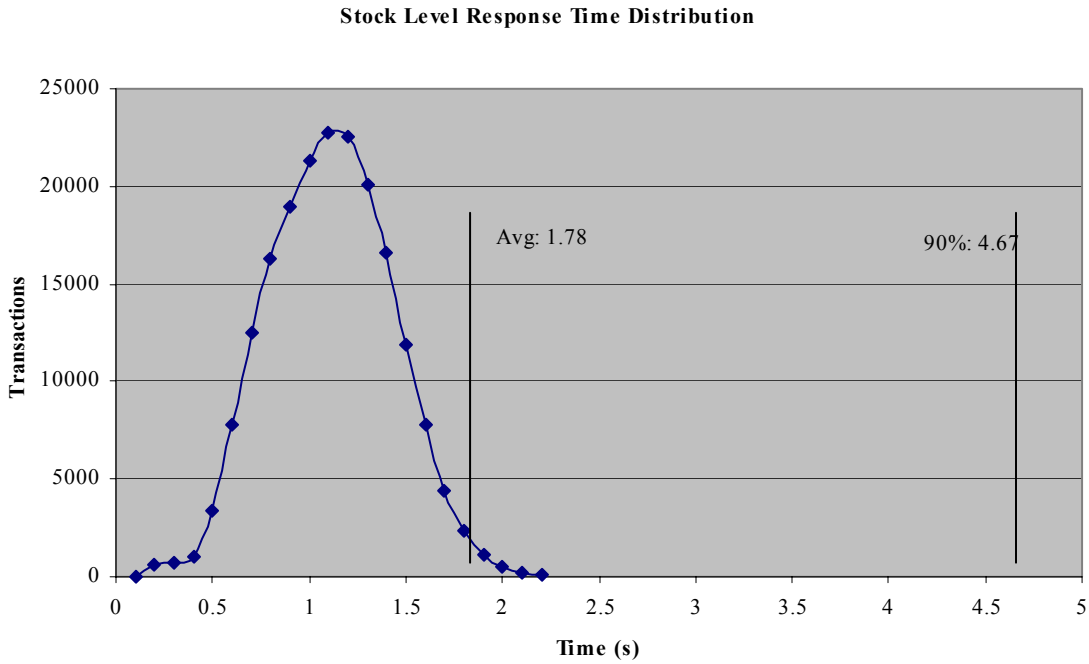


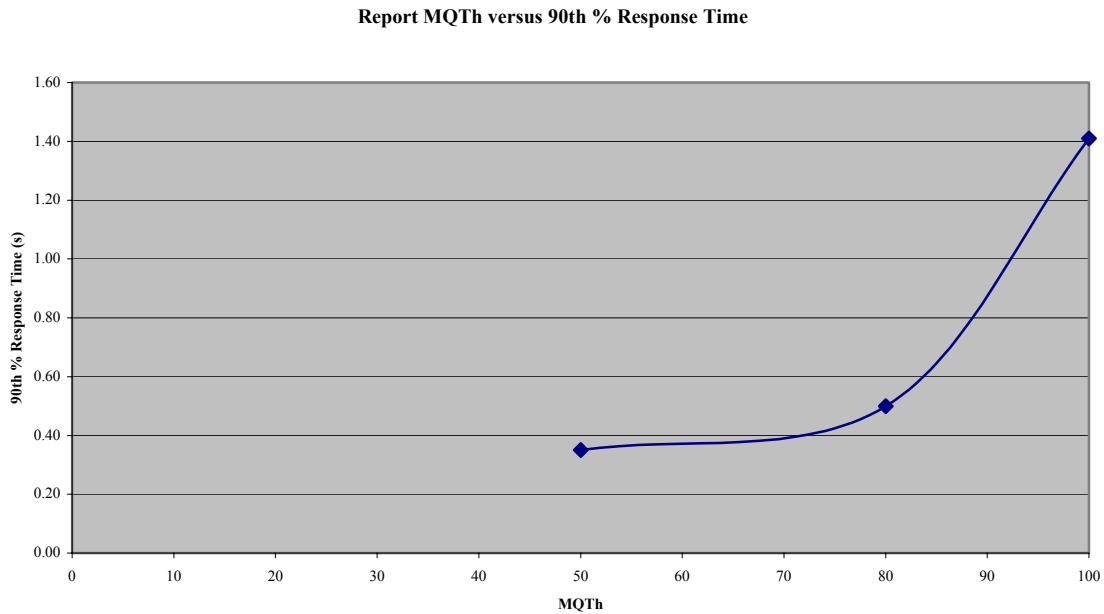
Figure 7: Stock Level Response Time Distribution



New-Order Response Time vs. Throughput Graph

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

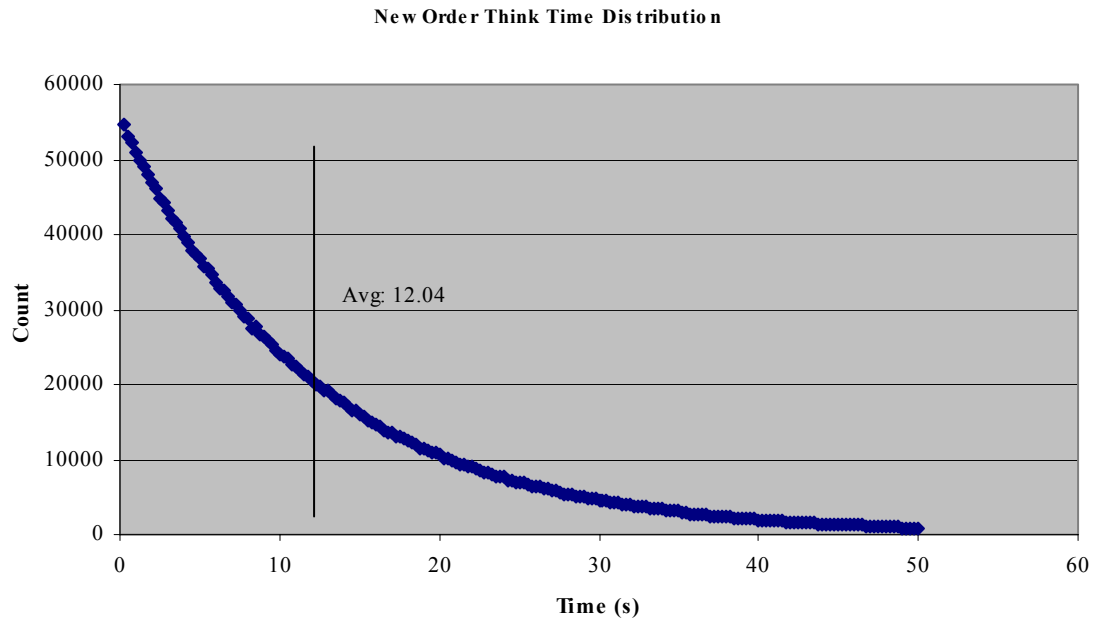
Figure 8: New Order Response Time vs. Throughput



New-Order Think Time Distribution Graph

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

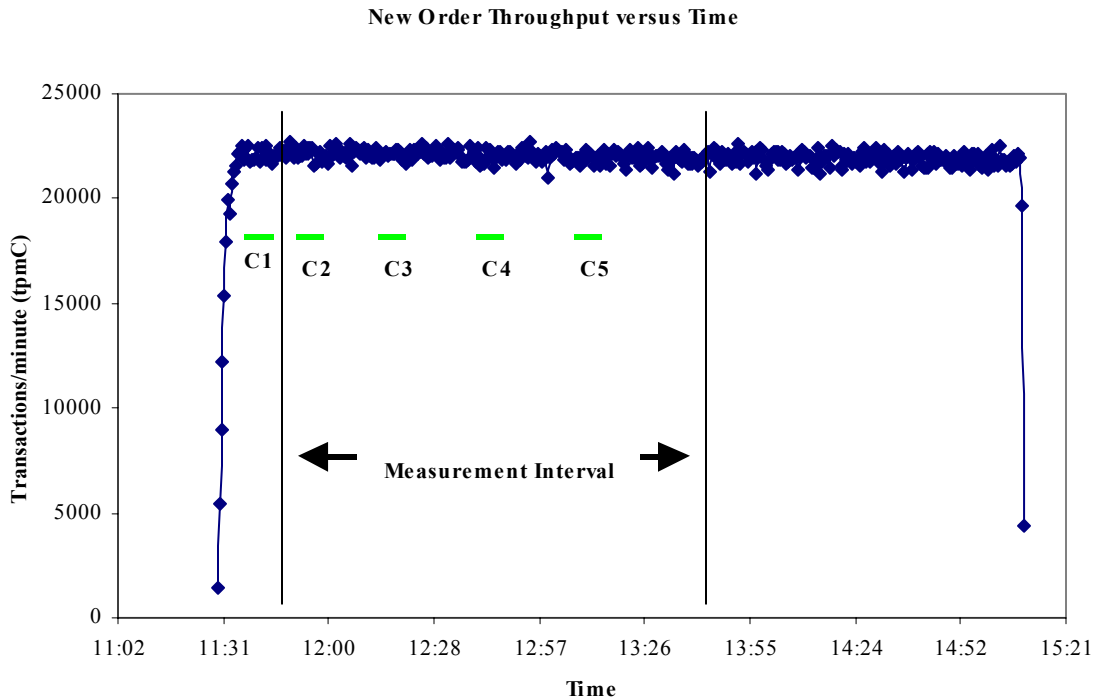
Figure 9: New Order Think Time Distribution



Steady-State Graph

A graph of throughput versus elapsed time (see Clause 5.6.5) must be reported for the New-Order transaction. (8.1.6.8)

Figure 10: New Order Throughput vs. Time



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. (8.1.6.9)

Steady state was determined using real time monitor utilities from both the operating system and the RTE. Steady state was further confirmed by the throughput data collected during the run and graphed in Figure 10.

Work Performed During Steady State

A description of how the work normally performed during a sustained test (for example checkpointing, writing redo/undo log records, etc.), actually occurred during the measurement interval must be reported. (8.1.6.10)

The RTE generated the required input data to choose a transaction from the menu. This data was timestamped. The menu response for the requested transaction was verified and timestamped in the RTE log files.

The RTE generated the required input data for the chosen transaction. It waited to complete the minimum required key time before transmitting the HTTP request to the client. The transmission was timestamped. The return of the screen with the required response data was timestamped.

The difference between these two timestamps was the response time for that transaction and was logged in the RTE log.

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

The RTE transmissions were sent to the web-based application program running on the client machines through Ethernet LANs. These web clients managed the emulated web browser interface as well as all requests to the database on the server. The applications communicated with the database server over another Ethernet LAN using the COM+ transaction monitor and Microsoft SQL Server DBLIB library and RPC calls.

To perform checkpoints at specific intervals, we set SQL Server *recovery interval* to the maximum allowable value and wrote a script to schedule multiple checkpoints at specific intervals. By setting the TRACE FLAG #3502, SQL Server logged the checkpoint beginning and ending time in the ERRORLOG file. The script included a wait time between each checkpoint equal to the measurement interval, which was 30 minutes. The checkpoint script was started manually after the RTE had all users logged in and sending transactions.

At each checkpoint, Microsoft SQL Server wrote to disk all memory pages that had been updated but not yet physically written to disk. Upon completion of the checkpoint, Microsoft SQL Server wrote a special record to the recovery log to indicate that all disk operations had been satisfied to this point.

Measurement Interval

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

The measurement interval was 7200 minutes.

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)) (8.1.6.11)

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

	Start	End	Duration
Measurement Interval	11:43:14	13:43:14	7,200
1 st Checkpoint	11:44:14	11:46:43	149
2 nd Checkpoint	12:14:09	12:16:39	150
3 rd Checkpoint	12:44:04	12:46:36	152
4 th Checkpoint	13:13:59	13:16:32	153

Transaction Mix

8.1.6.13 The method of regulation of the transaction mix (e.g., card decks or weighted random distribution) must be described. If weighted distribution is used and the RTE adjusts the weights associated with each transaction type, the maximum adjustments to the weight from the initial value must be disclosed. (8.1.6.13)

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

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

Table 9: Transaction Mix

Transaction	Percentage
New Order	44.84%
Payment	43.02%
Order Status	4.04%
Delivery	4.05%
Stock Level	4.05%

Other Metrics

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

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

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

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

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

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

Table 10: Transaction Statistics

Transaction	Function	Value
New Order	Home Warehouse Items	99.00%
	Remote Warehouse Items	1.00%
	Rolled Back Transactions	0.99%
	Average Lines Per Order	10.00
Payment	Home Warehouse	84.99%
	Remote Warehouse	15.01%
	Non-Primary Key Access	60.02%
Order Status	Non-Primary Key Access	60.07%
Delivery	Skipped Transactions	0

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.

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 timestamped the SUT output data for post-processing of the reported metrics. No other functionality was included on the driver system.

Figures 1 & 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 with 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:

- 10 BaseT (10 Mbit/sec) network segments between the RTE/Emulated Users and the switch.
- 100 BaseT (100 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 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: Feb 18, 2004

Software Availability Date: Feb 18, 2004

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: 22,052 tpmC

Price Performance Metric: \$1.50

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 the United States of America.

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 2003 Server Standard Licenses.
- 1 Microsoft SQL Server 2000 Standard Edition License.
- 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. All third party quotations are included at the end of this report as Appendix E.

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 Rucker St.
San Francisco, CA 94129-0920
Phone: (415) 561-6272, fax 415-561 6120
www.tpc.org

or:

Dell
One Dell Way
Round Rock, TX 78682
Attention: Mike Molloy, Ph.D.



February 5, 2004

Mr. Kong Yang
Dell Computer Corporation
One Dell Way
Round Rock, TX 78682

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

Platform: Dell PowerEdge 2650
Database Manager: Microsoft SQL Server 2000 Standard Edition
Operating System: Microsoft Windows 2003 Standard Server Edition
Transaction Monitor: COM+

System Under Test: Dell PowerEdge 2650 with:				
CPU's	Memory	Disks (total)	90% Response	TpmC
1 Intel Xeon @ 3.2 Ghz	Main: 2.5 GB	56 @ 18.2GB 4 @ 36GB	1.41	22,052.81

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.
- The database was properly scaled with 1900 warehouses, of which 1800 were active during the measured interval.
- The ACID properties were successfully demonstrated.
- Log loss and system loss were demonstrated on the full database with 1800 active warehouses.
- Data loss durability was demonstrated on a subset of the SUT configured with a database properly populated for 180 warehouses.
- 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 days space calculation was verified.
- The controller cache for the log disks was disabled.
- The steady state portion of the test was 120 minutes.
- More than one checkpoint was taken before the measured interval opened.
- Four checkpoints were completed inside the measured interval.
- The system pricing was checked for major components and maintenance.
- Third party quotes were verified for compliance.

Auditor Notes: None

Sincerely,

A handwritten signature in cursive script that reads "Lorna Livingtree".

Lorna Livingtree
Auditor

Appendix A - Application Source Code

Appendix A - Application Source Code

tpcc.dll ISAPI DLL 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          4001
#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 free element or -1 if this entry in use.
    int w_id;
    //warehouse id assigned at welcome form
    int d_id;
    //district id assigned at welcome form

    int iSyncId;
    //synchronization id
    int iTickCount;
    //time of last access;

    CTPC_BASE *pTxn;
} CLIENTDATA, *PCLIENTDATA;

//This structure is used to define the operational interface for terminal id support
typedef struct _TERM
{
    int iNumEntries;
    //total allocated terminal array entries
    int iFreeList;
    //next available terminal array element or -1 if none
    int iMasterSyncId;
    //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,
```

Appendix A - Application Source Code

```
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_IID_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_TOO_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_szTextDetail = new char[strlen(szTextDetail)+1];
    strcpy(m_szTextDetail, szTextDetail);
    m_SystemErr = dwSystemErr;
    m_szErrorText = NULL;
};

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

WEBERROR m_Error;
char *m_szTextDetail; //
char *m_szErrorText;
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 APIENTRY 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);
```

Appendix A - Application Source Code

```
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
FILEOS 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"
```

Appendix A - Application Source Code

```
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

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

#ifndef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//

////////////////////////////////////
#endif // 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\txnolog\include\rtetime.h"
#include "..\..\common\txnolog\include\spinlock.h"
#include "..\..\common\txnolog\include\txnolog.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
```

Appendix A - Application Source Code

```
// 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 //used to log delivery transaction information *txnDelilog = NULL;

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

// configuration settings from registry
TPCCREGISTRYDATA Reg;

DWORD dwNumDeliveryThreads = 4;
CRITICAL_SECTION DelBuffCriticalSection; //critical section
for delivery transactions cache
DELIVERY_TRANSACTION *pDelBuff = 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
 * DWORD ul_reason_for_call reason for call
 * LPVOID lpReserved
 *
 * reserved for future use
 *
 * RETURNS: BOOL FALSE errors
occured 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 txn monitor
                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
                    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");
                }
            }
        }
    }
}
```

Appendix A - Application Source Code

```
);
    hLibInstanceTm = LoadLibrary( szDllName
ERR_LOADDLL_FAILED, szDllName, GetLastError() );
    if (hLibInstanceTm == NULL)
        throw new CWEBCLNT_ERR(
// 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");
        szDllName );
        hLibInstanceDb = LoadLibrary(
        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 if (Reg.eDB_Protocol == ODBC)
    {
        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() );
    }
}
if (dwNumDeliveryThreads)
{
    // for deferred delivery txns:
    hDoneEvent = CreateEvent( NULL, TRUE /*
manual reset */, FALSE /* initially not signalled */, NULL );
    InitializeCriticalSection(&DelBuffCriticalSection);
    hWorkerSemaphore = CreateSemaphore(
    NULL, 0, dwDelBuffSize, NULL );
    dwDelBuffFreeCount = dwDelBuffSize;
    InitJulianTime(NULL);
// create unique log file name based on
    SYSTEMTIME Time;
    GetLocalTime( &Time );
    wsprintf( szLogFile, "%sdelivery-
%2.2d%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 );
    }
```


Appendix A - Application Source Code

```
        }
        break;
    case DLL_PROCESS_DETACH:
        if (dwNumDeliveryThreads)
        {
            if (txnDelilog != NULL)
            {
                //write event into txn log
                txnDelilog->
                >WriteCtrlRecToLog(TXN_EVENT_STOP, szMyComputerName, sizeof(szMyComputerName));

                // This will do a clean
                CTxnLog *txnDelilogLocal =
                txnDelilog;
                delete txnDelilogLocal;

                delete [] pDeliHandles;
                delete [] pDelBuff;

                CloseHandle( hWorkerSemaphore );
                CloseHandle( hDoneEvent );

                DeleteCriticalSection(&DelBuffCriticalSection);
            }

            DeleteCriticalSection(&TermCriticalSection);

            if (hLibInstanceTm != NULL)
                FreeLibrary( hLibInstanceTm );
            hLibInstanceTm = NULL;

            if (hLibInstanceDb != NULL)
                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->lpszExtensionDesc, "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
```

Appendix A - Application Source Code

```

*
*      HSE_STATUS_SUCCESS_AND_KEEP_CONN      keep connect valid comment sent
*
* COMMENTS:      None
*/
DWORD WINAPI HttpExtensionProc(EXTENSION_CONTROL_BLOCK *pECB)
{
    int          iCmd, FormId, TermId, iSyncId;
    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;
            }
    }
}

```

```

szBuffer);

        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;
    }
    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);
        break;

    case 5:
        // 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();
}

```

Appendix A - Application Source Code

```
        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(txnDeliRec != 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. "
            "%s. Server=%s, User=%s, Password=%s,
            e->ErrorText(), Reg.szDbServer, Reg.szDbUser,
            Database=%s",
            Reg.szDbPassword, Reg.szDbName );
        WriteMessageToEventLog( szTmp );
        delete e;
    }
}

```

Appendix A - Application Source Code

```
        goto ErrorExit;
    }
    catch (...)
    {
        WriteMessageToEventLog(TEXT("Unhandled exception caught in
DeliveryWorkerThread."));
        goto ErrorExit;
    }

    while (TRUE)
    {
        try
        {
            //while delivery thread running, i.e. user has not
            // need to wait for multiple objects: program
            while (TRUE)
            {
                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.queue);

                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)
            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
        txnDeliRec.TxnStatus = e->ErrorType();
        if (txnDeliLog != NULL)
            txnDeliLog->WriteToLog(&txnDeliRec);

        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 = o_carrier_id;
        GetLocalTime(&(pDelBuff+dwDelBuffFreeIndex)->queue);

        dwDelBuffFreeCount--;
        dwDelBuffFreeIndex++;
        if (dwDelBuffFreeIndex == dwDelBuffSize)
            dwDelBuffFreeIndex = 0;
        // wrap-around if
        // at end of buffer
    }
}
```

Appendix A - Application Source Code

```
    }
    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. CMD=command
    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, TERMIID, and SYNCID
    *pFormId = GetIntKeyValue(&ptr, "FORMID", NO_ERR, NO_ERR);
    *pTermId = GetIntKeyValue(&ptr, "TERMIID", 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=\"TERMIID\" VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\"
NAME=\"SYNCID\" VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\"
NAME=\"VERSION\" VALUE=\"\" WEBCLIENT_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>"
        "Max Connections =
<B>%d</B><BR>"
        "# of Delivery Threads =
<B>%d</B><BR>"
        "Max Pending Deliveries =
<B>%d</B><BR>"
        , szTxnMonNames[Reg.eTxnMon], szDBNames[Reg.eDB_Protocol],
        Reg.dwMaxConnections, dwNumDeliveryThreads, dwDelBuffSize
    );

    strcat( szBuffer, szTmp);
```

Appendix A - Application Source Code

```
if (Reg.eTxnMon == COM)
{
    sprintf( 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
sprintf( szTmp, "Please enter your database options for this
connection:<BR>"
        "DB Server = <INPUT
        "DB User ID = <INPUT
        "DB Password = <INPUT
        "DB Name = <INPUT
        , 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
sprintf( szTmp, "Database options which will be used by the
transaction monitor:<BR>"
        "DB Server =
        "DB User ID =
        "DB Password =
        "DB Name =
        , Reg.szDbServer, Reg.szDbUser,
        Reg.szDbPassword, Reg.szDbName );
    strcat( szBuffer, szTmp);

    sprintf( szTmp, "Please enter your Warehouse and District for this
session:<BR>"
            "Warehouse ID = <INPUT NAME=\"w_id\" SIZE=6><BR>"
            "District ID = <INPUT
NAME=\"d_id\" SIZE=2><BR>"
            "<PRE></font><HR>"
            "<INPUT TYPE=\"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 iNewTerm;
    char *ptr = pECB->lpszQueryString;

    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 CWBCLNT_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 CWBCLNT_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 CWBCLNT_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)
```

Appendix A - Application Source Code

```

        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 iTotl;

    EnterCriticalSection(&TermCriticalSection);

    iTotl = 0;
    for(i=0; i<Term.iNumEntries; i++)
    {
        if (Term.pClientData[i].iNextFree == -1)
            iTotl++;
    }

    LeaveCriticalSection(&TermCriticalSection);

    wsprintf( szBuffer,
        "<HTML><HEAD><TITLE>TPC-C Web Client Stats</TITLE></HEAD>"
        "<BODY><B><BIG> Total Active Connections: %d
</BIG></B><BR></BODY></HTML>"
        , iTotl );
}

char *CWEBCLNT_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_TERMINID,
        "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."
        },
    }
}

```

Appendix A - Application Source Code

```

    {
        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_TOO_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          *pQueryString      http string from
                client browser

```


Appendix A - Application Source Code

```
*
*   key value to look for      char      *pKey
*   character array into which to place key's value
*   maximum length of key value array.      int      iMax
*   error value to throw      WEBERROR      err
* RETURNS:      nothing.
* ERROR:      if (the pKey value is not found) then
*             if (err == 0)
*             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      *pQueryString      http string from
client browser
*
*               char      *pKey
*
*               WEBERROR      NoKeyErr      error
value to throw if key not found
```

```
*
*   value to throw if value not numeric      WEBERROR      NotIntErr      error
* RETURNS:      integer
* ERROR:      if (the pKey value is not found) then
*             if (NoKeyErr != NO_ERR)
*             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 != '&' && 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.
```

Appendix A - Application Source Code

```
*
*/
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
*
*/
```

Appendix A - Application Source Code

```
* 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.dll\" 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=\"TERMIN\" 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.dll\" 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=\"TERMIN\" 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.dll\" 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=\"TERMIN\" VALUE=\"%d\">"
"<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">"
"<PRE><font face=\"Courier\">
Level<BR>"
"Warehouse: %6.6d District: %2.2d<BR><BR>",
STOCK_LEVEL_FORM, iTermId, Term.pClientData[iTermId].iSyncId,
Term.pClientData[iTermId].w_id, Term.pClientData[iTermId].d_id);

    if ( bInput )
    {
        strcpy(szForm+c,
"Stock Level Threshold: <INPUT NAME=\"TT\" SIZE=2><BR>"
"low stock:    </font><BR> <BR> <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> <BR></PRE><HR>"
"<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..NewOrder..\">"

```

Appendix A - Application Source Code

```
VALUE=\".Payment..\">\"
    <INPUT TYPE=\"submit\" NAME=\"CMD\"
VALUE=\".Delivery..\">\"
    <INPUT TYPE=\"submit\" NAME=\"CMD\"
Status..\">\"
    <INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-
Level..\">\"
    <INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-
\"<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Exit..\">\"
\"</FORM></HTML>\"
, pStockLevelData->threshold, 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><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.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=\"TERMINID\" VALUE=\"%d\">\"
        \"<INPUT TYPE=\"hidden\" NAME=\"SYNCHID\" 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 += wsprintf(szForm+c, \"Warehouse: %6.6d \",
Term.pClientData[iTermId].w_id );
        strcpy( szForm+c,
            \"District: <INPUT NAME=\"DID\" SIZE=1>
            \"Customer: <INPUT NAME=\"CID\" SIZE=4> Name:
            \"Order Number: Number of Lines: W_tax:
            \" Supp_W Item_Id Item Name Qty Stock
B/G Price Amount<BR>\"
```

```

SIZE=6>
    <INPUT NAME=\"SP00*\" SIZE=4> <INPUT NAME=\"IID00*\"
    <INPUT NAME=\"Qty00*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP01*\" SIZE=4> <INPUT NAME=\"IID01*\"
    <INPUT NAME=\"Qty01*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP02*\" SIZE=4> <INPUT NAME=\"IID02*\"
    <INPUT NAME=\"Qty02*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP03*\" SIZE=4> <INPUT NAME=\"IID03*\"
    <INPUT NAME=\"Qty03*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP04*\" SIZE=4> <INPUT NAME=\"IID04*\"
    <INPUT NAME=\"Qty04*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP05*\" SIZE=4> <INPUT NAME=\"IID05*\"
    <INPUT NAME=\"Qty05*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP06*\" SIZE=4> <INPUT NAME=\"IID06*\"
    <INPUT NAME=\"Qty06*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP07*\" SIZE=4> <INPUT NAME=\"IID07*\"
    <INPUT NAME=\"Qty07*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP08*\" SIZE=4> <INPUT NAME=\"IID08*\"
    <INPUT NAME=\"Qty08*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP09*\" SIZE=4> <INPUT NAME=\"IID09*\"
    <INPUT NAME=\"Qty09*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP10*\" SIZE=4> <INPUT NAME=\"IID10*\"
    <INPUT NAME=\"Qty10*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP11*\" SIZE=4> <INPUT NAME=\"IID11*\"
    <INPUT NAME=\"Qty11*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP12*\" SIZE=4> <INPUT NAME=\"IID12*\"
    <INPUT NAME=\"Qty12*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP13*\" SIZE=4> <INPUT NAME=\"IID13*\"
    <INPUT NAME=\"Qty13*\" SIZE=1><BR>\"
SIZE=6>
    <INPUT NAME=\"SP14*\" SIZE=4> <INPUT NAME=\"IID14*\"
    <INPUT NAME=\"Qty14*\" SIZE=1><BR>\"
\"Execution Status:
Total:<BR>\"
}
else
{
    c += wsprintf(szForm+c, \"Warehouse: %6.6d District: %2.2d
Date: \",
        pNewOrderData->w_id,
        pNewOrderData->d_id);
    if ( bValid )
    {
        c += wsprintf(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 += wsprintf(szForm+c, \"<BR>Customer: %4.4d Name: %-16s Credit:
%-2s \",
        pNewOrderData->c_id, pNewOrderData->c_last, pNewOrderData-
>c_credit);
    if ( bValid )
```

Appendix A - Application Source Code

```

        {
            c += sprintf(szForm+c,
                "%Disc: %5.2f
Order Number: %8.8d Number
of Lines: %2.2d      W_tax: %5.2f  D_tax: %5.2f <BR> <BR>"
                " Supp_W Item_Id Item Name
Qty Stock B/G Price  Amount<BR>",
                100.0*pNewOrderData->c_discount,
                pNewOrderData->o_id,
                pNewOrderData->o_ol_cnt,
                100.0 * pNewOrderData->w_tax,
                100.0 * pNewOrderData->d_tax);

            for(i=0; i<pNewOrderData->o_ol_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:
                    " Supp_W Item_Id Item Name Qty
                    , pNewOrderData->o_id);

                i = 0;

                strncpy( szForm+c, szBR, (15-i)*5 );
                c += (15-i)*5;

                if ( bValid )
                    c += sprintf(szForm+c, "Execution Status: Transaction
                    Total: %8.2f ",
                    pNewOrderData->total_amount);
                else
                    c += sprintf(szForm+c, "Execution Status: Item number is
                    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..\">"
                    "Status..\">"

```

```

Level..\">"
                "<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-
                "<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;

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

    Payment<BR>"

        "Date: "
        , PAYMENT_FORM, iTermId, Term.pClientData[iTermId].iSyncId);

    if ( !bInput )
    {
        c += sprintf(szForm+c, "%2.2d-%2.2d-%4.4d %2.2d:%2.2d:%2.2d",
            pPaymentData->h_date.day,
            pPaymentData->h_date.month,
            pPaymentData->h_date.year,
            pPaymentData->h_date.hour,
            pPaymentData->h_date.minute,
            pPaymentData->h_date.second);
    }

    if ( bInput )
    {
        c += sprintf(szForm+c,
            "<BR> <BR>Warehouse: %6.6d"
            " District: <INPUT NAME=\"DID\"
            SIZE=1><BR> <BR> <BR> <BR> <BR>"
            "Customer: <INPUT NAME=\"CID\" SIZE=4>"
            "Cust-Warehouse: <INPUT NAME=\"CWI\" SIZE=4> "
            "Cust-District: <INPUT NAME=\"CDI\" SIZE=1><BR>"
            "Name: <INPUT NAME=\"CLT\" SIZE=16>"

            Since:<BR>"
            "
            Credit:<BR>"
            "
            Disc:<BR>"
            "
            Phone:<BR> <BR>"
            "Amount Paid: $<INPUT NAME=\"HAM\" SIZE=7>"

        New Cust-Balance:<BR>"

```

Appendix A - Application Source Code

```

    "Credit Limit:<BR> <BR>Cust-Data: <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>"
    , Term.pClientData[iTermId].w_id );
}
else
{
    c += sprintf(szForm+c,
        "<BR> <BR>Warehouse: %6.6d           District:
%2.2d<BR>"
            "%-20s           %-20s<BR>"
            "%-20s           %-20s<BR>"
            "%-20s %-2s %5.5s-%4.4s           %-20s %-2s %5.5s-%4.4s<BR>"
        "<BR>"
            "Customer: %4.4d Cust-Warehouse: %6.6d Cust-District:
%2.2d<BR>"
            "Name:  %-16s %-2s %-16s           Since: %2.2d-%2.2d-
%4.4d<BR>"
            "           %-20s           Credit: %-2s<BR>"
            , Term.pClientData[iTermId].w_id, pPaymentData->d_id
            , pPaymentData->w_street_1, pPaymentData->d_street_1
            , pPaymentData->w_street_2, pPaymentData->d_street_2
            , pPaymentData->w_city, pPaymentData->w_state,
pPaymentData->w_zip, pPaymentData->w_zip+5
            , pPaymentData->d_city, pPaymentData->d_state,
pPaymentData->d_zip, pPaymentData->d_zip+5
            , pPaymentData->c_id, pPaymentData->c_w_id,
pPaymentData->c_d_id
            , pPaymentData->c_first, pPaymentData->c_middle,
pPaymentData->c_last
            , pPaymentData->c_since.day, pPaymentData->c_since.month,
pPaymentData->c_since.year
            , pPaymentData->c_street_1, pPaymentData->c_credit
            );
    c += sprintf(szForm+c,
        "           %-20s           %%Disc:  %5.2f<BR>",
        pPaymentData->c_street_2, 100.0*pPaymentData->c_discount);
    c += sprintf(szForm+c,
        "           %-20s %-2s %5.5s-%4.4s           Phone: %6.6s-%3.3s-
%3.3s-%4.4s<BR> <BR>",
        pPaymentData->c_city, pPaymentData->c_state, pPaymentData-
>c_zip, pPaymentData->c_zip+5,
        pPaymentData->c_phone, pPaymentData->c_phone+6,
        pPaymentData->c_phone+9, pPaymentData->c_phone+12 );
    c += sprintf(szForm+c,
        "Amount Paid:           $$7.2f           New Cust-Balance:
$$14.2f<BR>"
            "Credit Limit:  $$13.2f<BR> <BR>"
            , pPaymentData->h_amount, pPaymentData->c_balance
            , pPaymentData->c_credit_lim
            );
    if ( pPaymentData->c_credit[0] == 'B' && pPaymentData->c_credit[1] ==
'c' )
        c += sprintf(szForm+c,
            "Cust-Data: %50.50s<BR>           %-
50.50s<BR>           %-50.50s<BR>","

```

```

            pPaymentData->c_data, pPaymentData->
c_data+50, pPaymentData->c_data+100, pPaymentData->c_data+150 );
        else
            strcpy(szForm+c, "Cust-Data: <BR> <BR> <BR> <BR>");
        strcat(szForm,
            " <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..\">"
            );
    }
}
/* FUNCTION: MakeOrderStatusForm
 *
 * COMMENTS:         The internal client buffer is created when the terminal id is
                    assigned and should not
                    longer needed.
                    be freed except when the client terminal id is no
                    longer needed.
 */
void MakeOrderStatusForm(int iTermId, ORDER_STATUS_DATA *pOrderStatusData, BOOL bInput,
char *szForm)
{
    int i, c;
    static char szBR[] = " <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR> <BR>
<BR> <BR> <BR> <BR>";
    c = sprintf(szForm,
        "<HTML><HEAD><TITLE>TPC-C Order-Status</TITLE></HEAD><BODY>"
        "<FORM ACTION=\"tpcc.dll\" METHOD=\"GET\">"
        "<INPUT TYPE=\"hidden\" NAME=\"STATUSID\" VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\" NAME=\"ERROR\" VALUE=\"0\">"
        "<INPUT TYPE=\"hidden\" NAME=\"FORMID\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"TERMIN\" VALUE=\"%d\">"
        "<INPUT TYPE=\"hidden\" NAME=\"SYNCID\" VALUE=\"%d\">"
        "<PRE><font face=\"Courier\">
            Order-
Status<BR>"
            "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>"
            "Cust-Balance:<BR> <BR>"
            "Order-Number:           Entry-Date:
Carrier-Number:<BR>"
            "Supply-W           Item-Id           Qty           Amount           Delivery-
Date<BR> <BR> <BR> <BR> <BR>"

```

Appendix A - Application Source Code

```

" <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
{
    c += sprintf(szForm+c,
        "District: %2.2d<BR>"
        "Customer: %4.4d Name: %-16s %-2s %-16s<BR>",
        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 Entry-Date: %2.2d-%2.2d-%4.4d
%2.2d:%2.2d:%2.2d Carrier-Number: %2.2d<BR>"
        "Supply-W Item-Id Qty Amount Delivery-
Date<BR>",
        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_ol_cnt; i++)
    {
        c += sprintf(szForm+c, " %6.6d %6.6d %2.2d
%8.2f %2.2d-%2.2d-%4.4d<BR>",
            pOrderStatusData->OL[i].ol_supply_w_id,
            pOrderStatusData->OL[i].ol_i_id,
            pOrderStatusData->OL[i].ol_quantity,
            pOrderStatusData->OL[i].ol_amount,
            pOrderStatusData->OL[i].ol_delivery_d.day,
            pOrderStatusData->OL[i].ol_delivery_d.month,
            pOrderStatusData->OL[i].ol_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=\\"TERMINID\\" VALUE=\\"%d\\">"
        "<INPUT TYPE=\\"hidden\\" NAME=\\"SYNCRID\\" 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>"
            "<INPUT TYPE=\\"submit\\" NAME=\\"CMD\\" VALUE=\\"Process\\">"
            "<INPUT TYPE=\\"submit\\" NAME=\\"CMD\\" VALUE=\\"Menu\\">"
            "</BODY></FORM></HTML>" );
    }
    else
    {
        sprintf( szForm+c,
            "Carrier Number: %2.2d<BR> <BR>"
            "Execution Status: %s <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=\\"..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,

```

Appendix A - Application Source Code

```
(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 );
}
```


Appendix A - Application Source Code

```
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 CWBCLNT_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 lpszQueryString    client
 *               browser http command string
 *               NEW_ORDER_DATA *pNewOrderData
 *               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] =
    { "SP00*", "SP01*", "SP02*", "SP03*", "SP04*",
      "SP05*", "SP06*", "SP07*", "SP08*", "SP09*",
      "SP10*", "SP11*", "SP12*", "SP13*", "SP14*" };
    static char szIID[MAX_OL_NEW_ORDER_ITEMS][7] =
    { "IID00*", "IID01*", "IID02*", "IID03*", "IID04*",
      "IID05*", "IID06*", "IID07*", "IID08*", "IID09*",
      "IID10*", "IID11*", "IID12*", "IID13*", "IID14*" };
    static char szQty[MAX_OL_NEW_ORDER_ITEMS][7] =
    { "Qty00*", "Qty01*", "Qty02*", "Qty03*", "Qty04*",
      "Qty05*", "Qty06*", "Qty07*", "Qty08*", "Qty09*",
      "Qty10*", "Qty11*", "Qty12*", "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 CWBCLNT_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 CWBCLNT_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 CWBCLNT_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 CWBCLNT_ERR(
ERR_NEWORDER_ITEMID_WITHOUT_SUPPW );
        }
    }
}
```

Appendix A - Application Source Code

```
        GetKeyValue(&ptr, szQty[i], szTmp, sizeof(szTmp),
ERR_NEWORDER_MISSING_QTY_KEY);
        if ( szTmp[0] )
            throw new CWBCLNT_ERR(
ERR_NEWORDER_QTY_WITHOUT_SUPPW );
    }
    if ( items == 0 )
        throw new CWBCLNT_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                lpszQueryString        client
browser http command string
 *                PAYMENT_DATA        *pPaymentData
 *                pointer to 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 CWBCLNT_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 CWBCLNT_ERR( ERR_PAYMENT_MISSING_CID_CLT );

        _strupr( szTmp );
        if ( strlen(pPaymentData->c_last) > LAST_NAME_LEN )
            throw new CWBCLNT_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 CWBCLNT_ERR( ERR_PAYMENT_CID_AND_CLT );
    }

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

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

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

    pOrderStatusData->d_id = GetIntKeyValue(&ptr, "DID*",
ERR_ORDERSTATUS_MISSING_DID_KEY, ERR_ORDERSTATUS_DID_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 CWBCLNT_ERR( ERR_ORDERSTATUS_MISSING_CID_CLT );

        _strupr( szTmp );
        if ( strlen(pOrderStatusData->c_last) > LAST_NAME_LEN )
            throw new CWBCLNT_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 CWBCLNT_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 CWBCLNT_ERR( ERR_ORDERSTATUS_CID_AND_CLT );
    }
}

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

```

Appendix A - Application Source Code

```
* 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 *dotpstr;
    BOOL bValid;

    if ( *ptr == 0 )
        return FALSE;

    // find decimal point
    dotpstr = strchr( ptr, '.' );
    if (dotpstr == NULL)
        // no decimal point, so just check for numeric
        return IsNumeric(ptr);
    *dotpstr = 0; // temporarily replace decimal with a terminator

    if ( *ptr != 0 )
        bValid = IsNumeric(ptr);
    // string starts with decimal point
    else if (*(dotpstr+1) == 0)
        return FALSE; // nothing but a decimal point is bad
    else
        bValid = TRUE;

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

    *dotpstr = '.'; // 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
#ifdef 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];
```

Appendix A - Application Source Code

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

Appendix A - Application Source Code

```
    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.dll 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

#ifdef _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          = 1,
    ERR_WARNING_LEVEL       = 2,
    ERR_INFORMATION_LEVEL   = 3
} 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_DELISTRV 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   //error from TPCW user class
#define ERR_TYPE_TPCW_ENG_OS 54
```

Appendix A - Application Source Code

```
#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());

    j += wsprintf(szTmp+j, "%s\n", ErrorText());

    ::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);
};
```

Appendix A - Application Source Code

```
~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);
        ErrorType() { return ERR_TYPE_OS;};
    };

    int ErrorType();
    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,LPTSTR);

    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
```

Appendix A - Application Source Code

```

#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 dblink, 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
{
    short                /* SQLSMALLINT */    year;
    unsigned short       /* SQLUSMALLINT */    month;
    unsigned short       /* SQLUSMALLINT */    day;
    unsigned short       /* SQLUSMALLINT */    hour;
    unsigned short       /* SQLUSMALLINT */    minute;
    unsigned short       /* SQLUSMALLINT */    second;
    unsigned long        /* SQLINTEGER */      fraction;
} TIMESTAMP_STRUCT;
#endif

// possible values for exec_status_code after transaction completes
enum EXEC_STATUS
{
    eOK,                // 0    "Transaction committed."
    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[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;

```


Appendix A - Application Source Code

```
        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;
    TIMESTAMP_STRUCT o_entry_d;
    short       o_carrier_id;
    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
    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 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

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

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

    = 0;    virtual PNEW_ORDER_DATA          BuffAddr_NewOrder()
    = 0;    virtual PPAYMENT_DATA          BuffAddr_Payment()
    = 0;    virtual PDELIVERY_DATA         BuffAddr_Delivery()

    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 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:  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
```

Appendix A - Application Source Code

```
*          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 <sqlldb.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;           // how many retries on
deadlock
static long     iConnectionCount = 0; // number of current dblib connections

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

BOOL WINAPI 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
            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              msgno
*              int                 msgstate
*              int                 severity
*              char                 *msgtext
*              printable message description
*
* RETURNS:      int                 INT_CONTINUE
                continue if error is SQLETIME else INT_CANCEL action
*
* INT_CANCEL    cancel operation
*
* COMMENTS:     This function also sets the dead lock dbproc variable if necessary.
*
*/

// 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 msgtext, 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 pDst and places
a
*              null character at the end of the destination string.
```

Appendix A - Application Source Code

```
*
* ARGUMENTS:      char          *pDest  destination string pointer
*                char          *pSrc   source
string pointer
*                int           n
*                number of characters to copy
*
* RETURNS:        None
*
* COMMENTS:       Unlike strncpy this function ensures that the result string is
*                always null terminated.
*/

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,          "Wrong version of stored
procs on database server" },
        { ERR_INVALID_CUST,              "Invalid Customer id,name."
customer." },
        { ERR_NO_SUCH_ORDER,             "No orders found for
succeeded." },
        { ERR_RETRIED_TRANS,             "Retries before transaction
"},
        { 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
    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
{
    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 (dbprocmsghandle(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 dblib 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);
```

Appendix A - Application Source Code

```
// 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 )
{
    // close db connection and deallocate resources
    dbclose(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
    {

```

Appendix A - Application Source Code

```
        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          iRowsRead = 0;
    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          iResultsRead = 0;
    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;
        }
    }
}
```

```
        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 tinyint
            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) != SUCCEED)
                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;
        }
        catch (CSQLERR *e)
        {
            if ((e->m_msgno == 1205 ||
                (e->m_msgno == iErrOleDbProvider &&
                 strstr(e->m_msgtext, sErrTimeoutExpired) !=
                 NULL)) &&
                (++iTryCount <= iMaxRetries))
            {
                // hit deadlock; backoff for increasingly longer
                delete e;
                Sleep(10 * iTryCount);
            }
        }
    }
}
```

Appendix A - Application Source Code

```
                else                throw;
            }
        } // while (TRUE)

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

void CTPCC_DBLIB::NewOrder()
{
    int                i;
    DBINT              commit_flag;
    DBDATETIME         datetime;
    DBDATERECC         daterec;

    int                iTryCount = 0;
    const BYTE         *pData;

    ResetError();

    while (TRUE)
    {
        try
        {
            dbrpcinit(m_dbproc, "tpcc_neworder", 0);

            dbrpcparam(m_dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.NewOrder.w_id);
            dbrpcparam(m_dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.NewOrder.d_id);
            dbrpcparam(m_dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.NewOrder.c_id);
            dbrpcparam(m_dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.NewOrder.o_ol_cnt);

            // check whether any order lines are for a remote warehouse
            m_txn.NewOrder.o_all_local = 1;
            for (i = 0; i < m_txn.NewOrder.o_ol_cnt; i++)
            {
                if (m_txn.NewOrder.OL[i].ol_supply_w_id !=
m_txn.NewOrder.w_id)
                {
                    m_txn.NewOrder.o_all_local = 0; // at
least one remote warehouse
                    break;
                }
            }
            dbrpcparam(m_dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.NewOrder.o_all_local);

            for (i = 0; i < m_txn.NewOrder.o_ol_cnt; i++)
            {
                dbrpcparam(m_dbproc, NULL, 0, SQLINT4, -1, -1,
(BYTE *) &m_txn.NewOrder.OL[i].ol_i_id);
                dbrpcparam(m_dbproc, NULL, 0, SQLINT4, -1, -1,
(BYTE *) &m_txn.NewOrder.OL[i].ol_supply_w_id);
                dbrpcparam(m_dbproc, NULL, 0, SQLINT2, -1, -1,
(BYTE *) &m_txn.NewOrder.OL[i].ol_quantity);
            }
        }
    }
}
```

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

// Get order line results
m_txn.NewOrder.total_amount = 0;
for (i = 0; i < m_txn.NewOrder.o_ol_cnt; i++)
{
    if (dbresults(m_dbproc) != SUCCEED)
        ThrowError(CDBLIBERR::eDbResults);

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

    if (dbnextrow(m_dbproc) != REG_ROW)
        ThrowError(CDBLIBERR::eDbNextRow);

    if (pData=dbdata(m_dbproc, 1))

        UtilStrCpy(m_txn.NewOrder.OL[i].ol_i_name, pData, dbdatlen(m_dbproc, 1));
    if (pData=dbdata(m_dbproc, 2))
        m_txn.NewOrder.OL[i].ol_stock =
(* (DBSMALLINT *) pData);
    if (pData=dbdata(m_dbproc, 3))

        UtilStrCpy(m_txn.NewOrder.OL[i].ol_brand_generic, pData, dbdatlen(m_dbproc,
3));
    if (pData=dbdata(m_dbproc, 4))
        dbconvert(m_dbproc, SQLNUMERIC,
(LPCBYTE)pData, dbdatlen(m_dbproc,4),
SQLFLT8, (BYTE
*)&m_txn.NewOrder.OL[i].ol_i_price, 8);
    if (pData=dbdata(m_dbproc, 5))
        dbconvert(m_dbproc, SQLNUMERIC,
(LPCBYTE)pData, dbdatlen(m_dbproc,5),
SQLFLT8, (BYTE
*)&m_txn.NewOrder.OL[i].ol_amount, 8);

    m_txn.NewOrder.total_amount =
m_txn.NewOrder.total_amount + m_txn.NewOrder.OL[i].ol_amount;

    DiscardNextRows(0);
}

// get remaining values for w_tax, d_tax, o_id, c_last,
c_discount, c_credit, o_entry_d, commit_flag
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))

    dbconvert(m_dbproc, SQLNUMERIC, (LPCBYTE)pData,
dbdatlen(m_dbproc,1), SQLFLT8, (BYTE *)&m_txn.NewOrder.w_tax, 8);
```

Appendix A - Application Source Code

```
        if (pData=dbdata(m_dbproc, 2))
            dbconvert(m_dbproc, SQLNUMERIC, (LPCBYTE)pData,
dbdatlen(m_dbproc,2), SQLFLT8, (BYTE *)&m_txn.NewOrder.d_tax, 8);
        if (pData=dbdata(m_dbproc, 3))
            m_txn.NewOrder.o_id = *(DBINT *) pData);
        if (pData=dbdata(m_dbproc, 4))
            UtilStrCpy(m_txn.NewOrder.c_last, pData,
dbdatlen(m_dbproc, 4));
        if (pData=dbdata(m_dbproc, 5))
            dbconvert(m_dbproc, SQLNUMERIC, (LPCBYTE)pData,
dbdatlen(m_dbproc,5), SQLFLT8, (BYTE *)&m_txn.NewOrder.c_discount, 8);
        if (pData=dbdata(m_dbproc, 6))
            UtilStrCpy(m_txn.NewOrder.c_credit, pData,
dbdatlen(m_dbproc, 6));
        if (pData=dbdata(m_dbproc, 7))
        {
            datetime = *((DBDATETIME *) pData);
            dbdatecrack(m_dbproc, &daterec, &datetime);
            m_txn.NewOrder.o_entry_d.year = daterec.year;
            m_txn.NewOrder.o_entry_d.month = daterec.month;
            m_txn.NewOrder.o_entry_d.day = daterec.day;
            m_txn.NewOrder.o_entry_d.hour = daterec.hour;
            m_txn.NewOrder.o_entry_d.minute = daterec.minute;
            m_txn.NewOrder.o_entry_d.second = daterec.second;
        }
        if (pData=dbdata(m_dbproc, 8))
            commit_flag = *(DBTINYINT *) pData);

        DiscardNextRows(0);
        DiscardNextResults(0);

        if (commit_flag == 1)
        {
            m_txn.NewOrder.total_amount *= ((1 +
m_txn.NewOrder.w_tax + m_txn.NewOrder.d_tax) * (1 - m_txn.NewOrder.c_discount));
            m_txn.NewOrder.exec_status_code = eOK;
        }
        else
            m_txn.NewOrder.exec_status_code = eInvalidItem;

        return;
    }
    catch (CSQLERR *e)
    {
        if ((e->m_msgno == 1205 ||
(e->m_msgno == iErrOleDbProvider &&
strstr(e->m_msgtext, sErrTimeoutExpired) !=
NULL)) &&
        {
            (++iTryCount <= iMaxRetries)
            {
                // hit deadlock; backoff for increasingly longer
                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::Payment()
{
    DBDATETIME datetime;
    DBDATEREC daterec;

    int iTryCount = 0;
    const BYTE *pData;

    ResetError();

    while (TRUE)
    {
        try
        {
            dbrpcinit(m_dbproc, "tpcc_payment", 0);

            dbrpcparam(m_dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.Payment.w_id);
            dbrpcparam(m_dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.Payment.c_w_id);
            dbrpcparam(m_dbproc, NULL, 0, SQLFLT8, -1, -1, (BYTE *)
&m_txn.Payment.h_amount);
            dbrpcparam(m_dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.Payment.d_id);
            dbrpcparam(m_dbproc, NULL, 0, SQLINT1, -1, -1, (BYTE *)
&m_txn.Payment.c_d_id);
            dbrpcparam(m_dbproc, NULL, 0, SQLINT4, -1, -1, (BYTE *)
&m_txn.Payment.c_id);

            // if customer id is zero, then payment is by name
            if (m_txn.Payment.c_id == 0)
                dbrpcparam(m_dbproc, NULL, 0, SQLCHAR, -1,
strlen(m_txn.Payment.c_last), (unsigned char *)m_txn.Payment.c_last);

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

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

            if (dbnextrow(m_dbproc) != REG_ROW)
                ThrowError(CDBLIBERR::eDbNextRow);

            if (dbnumcols(m_dbproc) != 27)
                ThrowError(CDBLIBERR::eWrongNumCols);

            if (pData=dbdata(m_dbproc, 1))
                m_txn.Payment.c_id = *(DBINT *) pData);
            if (pData=dbdata(m_dbproc, 2))
                UtilStrCpy(m_txn.Payment.c_last, pData,
dbdatlen(m_dbproc, 2));

            if (pData=dbdata(m_dbproc, 3))
            {
                datetime = *((DBDATETIME *) pData);
                dbdatecrack(m_dbproc, &daterec, &datetime);
                m_txn.Payment.h_date.year = daterec.year;
                m_txn.Payment.h_date.month = daterec.month;
                m_txn.Payment.h_date.day = daterec.day;
            }
        }
        catch (CSQLERR *e)
        {
            if ((e->m_msgno == 1205 ||
(e->m_msgno == iErrOleDbProvider &&
strstr(e->m_msgtext, sErrTimeoutExpired) !=
NULL)) &&
            {
                (++iTryCount <= iMaxRetries)
                {
                    // hit deadlock; backoff for increasingly longer
                    delete e;
                    Sleep(10 * iTryCount);
                }
                else
                    throw;
            }
        }
    }
} // while (TRUE)

// if (iTryCount)
```

Appendix A - Application Source Code

```
        m_txn.Payment.h_date.hour = daterec.hour;
        m_txn.Payment.h_date.minute = daterec.minute;
        m_txn.Payment.h_date.second = daterec.second;
    }
    if (pData=dbdata(m_dbproc, 4))
        UtilStrCpy(m_txn.Payment.w_street_1, pData,
dbdatlen(m_dbproc, 4));
    if (pData=dbdata(m_dbproc, 5))
        UtilStrCpy(m_txn.Payment.w_street_2, pData,
dbdatlen(m_dbproc, 5));
    if (pData=dbdata(m_dbproc, 6))
        UtilStrCpy(m_txn.Payment.w_city, pData,
dbdatlen(m_dbproc, 6));
    if (pData=dbdata(m_dbproc, 7))
        UtilStrCpy(m_txn.Payment.w_state, pData,
dbdatlen(m_dbproc, 7));
    if (pData=dbdata(m_dbproc, 8))
        UtilStrCpy(m_txn.Payment.w_zip, pData,
dbdatlen(m_dbproc, 8));
    if (pData=dbdata(m_dbproc, 9))
        UtilStrCpy(m_txn.Payment.d_street_1, pData,
dbdatlen(m_dbproc, 9));
    if (pData=dbdata(m_dbproc, 10))
        UtilStrCpy(m_txn.Payment.d_street_2, pData,
dbdatlen(m_dbproc, 10));
    if (pData=dbdata(m_dbproc, 11))
        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 == iErrOleDbProvider &&
strstr(e->m_msgtext, sErrTimeoutExpired) !=
NULL)) &&
(++iTryCount <= iMaxRetries))
    {
        // hit deadlock; backoff for increasingly longer
        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 i;
    DBDATETIME datetime;
```


Appendix A - Application Source Code

```
DBDATEREC daterec;

int          iTryCount = 0;
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 zero, then order status is by name
        if (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 = dbnextrow(m_dbproc);
            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   =
m_txn.OrderStatus.o_entry_d.month   =
m_txn.OrderStatus.o_entry_d.day    = daterec.day;
                m_txn.OrderStatus.o_entry_d.hour   =
m_txn.OrderStatus.o_entry_d.minute =
m_txn.OrderStatus.o_entry_d.second =
                daterec.second;
            }
        }
    }
}
```

Appendix A - Application Source Code

```
        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 == iErrOleDbProvider &&
strstr(e->m_msgtext, sErrTimeoutExpired) !=
NULL)) &&
            (++iTryCount <= iMaxRetries))
        {
            // hit deadlock; backoff for increasingly longer
            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 ||
(e->m_msgno == iErrOleDbProvider &&
strstr(e->m_msgtext, sErrTimeoutExpired) !=
NULL)) &&
            (++iTryCount <= iMaxRetries))
        {
            // hit deadlock; backoff for increasingly longer
            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;
    }
}
```

Appendix A - Application Source Code

```
    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

#ifndef 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.
#ifndef 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, // error from
        eLogin, // error from dbopen
        eDbOpen, // error from dbuse
        eDbUse, // error from
        eDbSqlExec, // error from
        eDbSet, // error from one
        eDbNextRow, // error from
        eWrongRowCount, // more or less rows returned
        eWrongNumCols, // more or less columns
        eDbResults, // error from
        eDbRpcExec, // error from
        eDbSetMaxProcs, // error from dbsetmaxprocs
        eDbProcHandler // error from either
    };

    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;};
};
```

Appendix A - Application Source Code

```
};

class CTPCC_DBLIB_ERR : public CBaseErr
{
public:
    enum CTPCC_DBLIB_ERRS
    {
        ERR_WRONG_SP_VERSION = 1,    // "Wrong version of stored
procs on database server"
        ERR_INVALID_CUST,            // "Invalid
Customer id,name."
        ERR_NO_SUCH_ORDER            // "No orders found
for customer."
    };

    CTPCC_DBLIB_ERR( int iErr ) { m_errno = iErr; };

    int m_errno;

    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_handler
// 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"
```

Appendix A - Application Source Code

```
#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_vTxn.parray);

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

void CTPCC_COM::NewOrder()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pNewOrder->NewOrder(m_vTxn, &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 );
```

Appendix A - Application Source Code

```
}
void CTPCC_COM::Payment()
{
    VARIANT vTxn_out;

    HRESULT hr = m_pPayment->Payment(m_vTxn, &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_vTxn, &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_vTxn, &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.
#ifndef 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;
    }
}
```

Appendix A - Application Source Code

```
};

}

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 NewOrder;
            PAYMENT_DATA Payment;
            DELIVERY_DATA Delivery;
            STOCK_LEVEL_DATA StockLevel;
            ORDER_STATUS_DATA 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_LOADDDL_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, char *szTextDetail, 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;
};
```

Appendix A - Application Source Code

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

CTPCC_Common();
~CTPCC_Common();

// ITPCC
public:
HRESULT __stdcall NewOrder(          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();

// IObjectControl
STDMETHODIMP CanBePooled() { return m_bCanBePooled; }
STDMETHODIMP Activate() { return S_OK; } // we don't support COM
Services transactions (no enlistment)
STDMETHODIMP Deactivate() { /* nothing to do */ }

// IObjectConstruct
STDMETHODIMP Construct(IDispatch * pUnk);

// helper methods
private:
BOOL          m_bCanBePooled;
CTPCC_BASE   *_m_pTxn;

struct COM_DATA
{
int retval;
int error;
union
{
NEW_ORDER_DATA          NewOrder;

```

```
PAYMENT_DATA          Payment;
DELIVERY_DATA         Delivery;
STOCK_LEVEL_DATA      StockLevel;
ORDER_STATUS_DATA    OrderStatus;
};

};

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

BEGIN_COM_MAP(CTPCC)
COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
END_COM_MAP()

};

////////////////////////////////////
// CNewOrder
class CNewOrder :
public CTPCC_Common,
public CComCoClass<CNewOrder, &CLSID_NewOrder>
{
public:
DECLARE_REGISTRY_RESOURCEID(IDR_NEWORDER)

BEGIN_COM_MAP(CNewOrder)
COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
END_COM_MAP()

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

////////////////////////////////////
// COrderStatus
class COrderStatus :
public CTPCC_Common,
public CComCoClass<COrderStatus, &CLSID_OrderStatus>
{
public:
DECLARE_REGISTRY_RESOURCEID(IDR_ORDERSTATUS)

BEGIN_COM_MAP(COrderStatus)

```


Appendix A - Application Source Code

```
COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
END_COM_MAP()

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

////////////////////////////////////
// CPayment
class CPayment :
    public CTPCC_Common,
    public CComCoClass<CPayment, &CLSID_Payment>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_PAYMENT)

    BEGIN_COM_MAP(CPayment)
        COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()

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

////////////////////////////////////
// CStockLevel
class CStockLevel :
    public CTPCC_Common,
    public CComCoClass<CStockLevel, &CLSID_StockLevel>
{
public:
    DECLARE_REGISTRY_RESOURCEID(IDR_STOCKLEVEL)

    BEGIN_COM_MAP(CStockLevel)
        COM_INTERFACE_ENTRY2(IUnknown, CComObjectRootEx)
        COM_INTERFACE_ENTRY_CHAIN(CTPCC_Common)
    END_COM_MAP()

    // ITPCC
    public:
        HRESULT __stdcall NewOrder(
            VARIANT txn_in, VARIANT* txn_out)
        {return E_NOTIMPL;}
```

```
        HRESULT __stdcall Payment(
            VARIANT txn_in, VARIANT* txn_out)
    {return E_NOTIMPL;}
    // HRESULT __stdcall StockLevel( VARIANT txn_in, VARIANT* txn_out) {return
E_NOTIMPL;}
    HRESULT __stdcall OrderStatus(
        VARIANT txn_in, VARIANT* txn_out)
    {return E_NOTIMPL;}
};
```

tpcc_com_all/src/resource.h

```
//{{NO_DEPENDENCIES}}
// Microsoft Developer Studio generated include file.
// Used by tpcc_com_all.rc
//
#define IDS_PROJNAME            100
#define IDR_TPCC                101
#define IDR_NEWORDER           102
#define IDR_ORDERSTATUS        103
#define IDR_PAYMENT            104
#define IDR_STOCKLEVEL         105

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifdef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE        202
#define _APS_NEXT_COMMAND_VALUE        32768
#define _APS_NEXT_CONTROL_VALUE        201
#define _APS_NEXT_SYMED_VALUE         106
#endif
#endif
```

tpcc_com_all/src/tpcc_com_all.cpp

```
/*      FILE:                TPCCOM_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>
```

Appendix A - Application Source Code

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

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

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

#include "tpcc_com_ps.h"
#include "..\..\common\src\trans.h" //tpckit
transaction header contains definitions 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;
}
```

Appendix A - Application Source Code

```
        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 DllGetClassObject(REFCLSID rclsid, REFIID riid, LPVOID* ppv)
    {
        return _Module.GetClassObject(rclsid, 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.
GetProcAddr 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)
```

Appendix A - Application Source Code

```
        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* pObjectContext = NULL;

    // get our object context
    HRESULT hr = CoGetObjectContext( IID_IObjectContext, (void **) &pObjectContext
);
    pObjectContext->SetComplete();
    ReleaseInterface(pObjectContext);
    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, szMyComputerName, Reg.szDbName );
        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));

        m_pTxn->NewOrder();           // do the actual txn

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector(VT_UI1,
txn_in.parray-
>rgsabound->cElements,
txn_in.parray-
>rgsabound->cElements);
        pData = (COM_DATA*) txn_out->parray->pvData;

        memcpy( &pData->u.NewOrder, pNewOrder, sizeof(NEW_ORDER_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        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)
||
10054)) )
            ((e->ErrorType() == ERR_TYPE_ODBC) && (e->ErrorNum() ==
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;
    }
}

HRESULT CTPCC_Common::Payment(VARIANT txn_in, VARIANT* txn_out)
{
    PPAYMENT_DATA    pPayment;

```

Appendix A - Application Source Code

```
COM_DATA          *pData;
try
{
    pData = (COM_DATA*)txn_in.parray->pvData;
    pPayment = m_pTxn->BuffAddr_Payment();

    memcpy(pPayment, &pData->u.Payment, sizeof(PAYMENT_DATA));

    m_pTxn->Payment();          // do the actual txn

    VariantInit(txn_out);
    txn_out->vt = VT_SAFEARRAY;
    txn_out->parray = SafeArrayCreateVector( VT_UI1,
>rgsabound->cElements,          txn_in.parray-
>rgsabound->cElements);          txn_in.parray-
    pData = (COM_DATA*) txn_out->parray->pvData;

    memcpy( &pData->u.Payment, pPayment, sizeof(PAYMENT_DATA));

    pData->retval = ERR_SUCCESS;
    pData->error = 0;
    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)
||
10054)) )
        ((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;
}
}

HRESULT CTPCC_Common::StockLevel(VARIANT txn_in, VARIANT* txn_out)
{
    PSTOCK_LEVEL_DATA  pStockLevel;
    COM_DATA            *pData;

    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pStockLevel = m_pTxn->BuffAddr_StockLevel();

        memcpy(pStockLevel, &pData->u.StockLevel, sizeof(STOCK_LEVEL_DATA));

        m_pTxn->StockLevel();
    }
}
```

```
VariantInit(txn_out);
txn_out->vt = VT_SAFEARRAY;
txn_out->parray = SafeArrayCreateVector( VT_UI1,
>rgsabound->cElements,          txn_in.parray-
>rgsabound->cElements);          txn_in.parray-
    pData = (COM_DATA*)txn_out->parray->pvData;

    memcpy( &pData->u.StockLevel, pStockLevel, sizeof(STOCK_LEVEL_DATA));

    pData->retval = ERR_SUCCESS;
    pData->error = 0;
    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)
||
10054)) )
        ((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;
}
}

HRESULT CTPCC_Common::OrderStatus(VARIANT txn_in, VARIANT* txn_out)
{
    PORDER_STATUS_DATA  pOrderStatus;
    COM_DATA            *pData;

    try
    {
        pData = (COM_DATA*)txn_in.parray->pvData;
        pOrderStatus = m_pTxn->BuffAddr_OrderStatus();

        memcpy(pOrderStatus, &pData->u.OrderStatus,
sizeof(ORDER_STATUS_DATA));

        m_pTxn->OrderStatus();

        VariantInit(txn_out);
        txn_out->vt = VT_SAFEARRAY;
        txn_out->parray = SafeArrayCreateVector( VT_UI1,
>rgsabound->cElements,          txn_in.parray-
>rgsabound->cElements);          txn_in.parray-
        pData = (COM_DATA*)txn_out->parray->pvData;
    }
}
```

Appendix A - Application Source Code

```
        memcpy( &pData->u.OrderStatus, pOrderStatus,
sizeof(ORDER_STATUS_DATA));

        pData->retval = ERR_SUCCESS;
        pData->error = 0;
        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 5.03.0280 */
/* at Sat Apr 08 16:40:18 2000
```

```
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
   Oicf (OptLev=i2), Wl, Zp8, env=Win32 (32b run), 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 tpcc_com_all_h
#define tpcc_com_all_h

/* Forward Declarations */

#ifndef __TPCC_FWD_DEFINED__
#define __TPCC_FWD_DEFINED__

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

#endif /* __TPCC_FWD_DEFINED__ */

#ifndef __NewOrder_FWD_DEFINED__
#define __NewOrder_FWD_DEFINED__

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

#endif /* __NewOrder_FWD_DEFINED__ */

#ifndef __OrderStatus_FWD_DEFINED__
#define __OrderStatus_FWD_DEFINED__

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

#endif /* __OrderStatus_FWD_DEFINED__ */

#ifndef __Payment_FWD_DEFINED__
#define __Payment_FWD_DEFINED__
```

Appendix A - Application Source Code

```
#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_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR * );

/* 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_TPCCLib;

#ifdef __cplusplus
class DECLSPEC_UUID("122A3128-2520-11D3-BA71-00C04FBFE08B")
TPCCLib;
#endif
#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
#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: TPCCLib.IDL
 * Microsoft TPC-C Kit Ver. 4.20.000
 * Copyright Microsoft, 1999
 * All Rights Reserved
 * not yet audited
 * PURPOSE: IDL source for TPCCLib.dll. This file is processed by the MIDL tool to
 * produce the type library (TPCCLib.tlb) and
 * marshalling code.
 */
```

Appendix A - Application Source 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)
#ifndef 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"

```


Appendix A - Application Source Code

```
END

#endif // 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
////////////////////////////////////

#ifdef APSTUDIO_INVOKED
////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//
1 TYPELIB "tpcc_com_all.tlb"

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

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 'TPCC.AllTxns.1'
            VersionIndependentProgID = s 'TPCC.AllTxns'
            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 */
```

Appendix A - Application Source Code

```
/* 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 5.03.0280 */
/* at Sat Apr 08 16:40:18 2000
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
Oicf (OptLev=i2), Wl, Zp8, env=Win32 (32b run), 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_AXP64)

#ifdef __cplusplus
extern "C"{
#endif

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

#ifdef _MIDL_USE_GUIDDEF_

#ifdef 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_

#ifdef __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__

#ifdef 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,0x122A3117,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_TPCC,0x122A3128,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_NewOrder,0x975BAABF,0x84A7,0x11D2,0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_OrderStatus,0x266836AD,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_Payment,0xCD02F7EF,0xA4FA,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_StockLevel,0x2668369E,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* !defined(_M_IA64) && !defined(_M_AXP64) */

#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 5.03.0280 */
/* at Sat Apr 08 16:40:18 2000
*/
/* Compiler settings for .\src\tpcc_com_all.idl:
Oicf (OptLev=i2), Wl, Zp8, env=Win64 (32b run,appending), ms_ext, c_ext, robust
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_AXP64)

#ifdef __cplusplus
extern "C"{
#endif

#include <rpc.h>
```

Appendix A - Application Source Code

```
#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,0x122A3117,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_TPCC,0x122A3128,0x2520,0x11D3,0xBA,0x71,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_NewOrder,0x975BAABF,0x84A7,0x11D2,0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_OrderStatus,0x266836AD,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_Payment,0xCD02F7EF,0xA4FA,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);

MIDL_DEFINE_GUID(CLSID,
CLSID_StockLevel,0x2668369E,0xA50D,0x11D2,0xBA,0x4E,0x00,0xC0,0x4F,0xBF,0xE0,0x8B);
```

```
#undef MIDL_DEFINE_GUID

#ifdef __cplusplus
}
#endif

#endif /* defined(_M_IA64) || defined(_M_AXP64)*/
```

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 'TPCC.NewOrder.1'
            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
    {
```

Appendix A - Application Source Code

```
ForceRemove {266836AD-A50D-11D2-BA4E-00C04FBFE08B} = s 'OrderStatus
Class'
{
    ProgID = s 'TPCC.OrderStatus.1'
    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 'TPCC.Payment.1'
        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 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), Wl, Zp8, env=Win32 (32b run), 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_

/* 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_FAR * __RPC_USER MIDL_user_allocate(size_t);
void __RPC_USER MIDL_user_free( void __RPC_FAR * );

/* 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;
```

Appendix A - Application Source Code

```
#if defined(__cplusplus) && !defined(CINTERFACE)

MIDL_INTERFACE("FEEB6AA2-84B1-11d2-BA47-00C04FBFE08B")
ITPCC : public IUnknown
{
public:
    virtual HRESULT __stdcall NewOrder(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

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

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

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

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

    virtual HRESULT __stdcall CallSetComplete( void) = 0;

};

#else /* C style interface */

typedef struct ITPCCVtbl
{
    BEGIN_INTERFACE

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

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

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

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

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

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

};

#endif
```

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

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

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

END_INTERFACE
} ITPCCVtbl;

interface ITPCC
{
    CONST_VTBL struct ITPCCVtbl __RPC_FAR *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 __RPC_FAR * This,
```

Appendix A - Application Source Code

```
/* [in] */ VARIANT txn_in,
/* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCC_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

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

void __RPC_STUB ITPCC_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

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

void __RPC_STUB ITPCC_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

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

void __RPC_STUB ITPCC_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);

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

void __RPC_STUB ITPCC_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer *pRpcChannelBuffer,
    PRPC_MESSAGE_pRpcMessage,
    DWORD *_pdwStubPhase);
```

```
HRESULT __stdcall ITPCC_CallSetComplete_Proxy(
    ITPCC __RPC_FAR * 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 __RPC_FAR *,
    unsigned long __RPC_FAR *, VARIANT __RPC_FAR * );
unsigned char __RPC_FAR * __RPC_USER VARIANT_UserMarshal( unsigned long __RPC_FAR *,
    unsigned char __RPC_FAR *, VARIANT __RPC_FAR * );
unsigned char __RPC_FAR * __RPC_USER VARIANT_UserUnmarshal( unsigned long __RPC_FAR *,
    unsigned char __RPC_FAR *, VARIANT __RPC_FAR * );
void __RPC_USER VARIANT_UserFree( unsigned long __RPC_FAR *,
    VARIANT __RPC_FAR * );

/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif

#endif
```

tpcc_com_all/src/tpcc_com_sl.rgs

```
HKCR
{
    TPCC.StockLevel.1 = s 'StockLevel Class'
    {
        CLSID = s '{2668369E-A50D-11D2-BA4E-00C04FBFE08B}'
    }
    TPCC.StockLevel = s 'StockLevel Class'
    {
        CurVer = s 'TPCC.StockLevel.1'
    }
    NoRemove CLSID
    {
        ForceRemove {2668369E-A50D-11D2-BA4E-00C04FBFE08B} = s 'StockLevel
Class'
    {
        ProgID = s 'TPCC.StockLevel.1'
        VersionIndependentProgID = s 'TPCC.StockLevel'
        InprocServer32 = s '%MODULE%'
    }
    }
```

Appendix A - Application Source Code

```
        val ThreadingModel = s 'Both'
    }
}
}
```

tpcc_com_ps/src/dlldata.c

```
/*
 * *****
 * DllData file -- generated by MIDL compiler
 *
 * DO NOT ALTER THIS FILE
 *
 * This file is regenerated by MIDL on every IDL file compile.
 *
 * To completely reconstruct this file, delete it and rerun MIDL
 * on all the IDL files in this DLL, specifying this file for the
 * /dlldata command line option
 *
 * *****
 */

#include <rpcproxy.h>

#ifdef __cplusplus
extern "C" {
#endif

EXTERN_PROXY_FILE( tpcc_com_ps )

PROXYFILE_LIST_START
/* Start of list */
REFERENCE_PROXY_FILE( tpcc_com_ps ),
/* End of list */
PROXYFILE_LIST_END

DLLDATA_ROUTINES( aProxyFileList, GET_DLL_CLSID )

#ifdef __cplusplus
} /*extern "C" */
#endif

/* end of generated dlldata file */
```

tpcc_com_ps/src/tpcc_com_ps.def

```
LIBRARY      "tpcc_com_ps"
DESCRIPTION  'Proxy/Stub DLL'
EXPORTS
    DllGetClassObject      @1    PRIVATE
```

```
DllCanUnloadNow      @2    PRIVATE
GetProxyDllInfo      @3    PRIVATE
DllRegisterServer    @4    PRIVATE
DllUnregisterServer   @5    PRIVATE
```

tpcc_com_ps/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 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000
 */
/* Compiler settings for .\src\tpcc_com_ps.idl:
   Oicf (OptLev=i2), Wl, Zp8, env=Win32 (32b run), 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 __RPCNDR_H_VERSION__
#error this stub requires an updated version of <rpcndr.h>
#endif // __RPCNDR_H_VERSION__

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

#ifdef __tpcc_com_ps_h__
#define __tpcc_com_ps_h__

/* Forward Declarations */

#ifdef __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"
```

Appendix A - Application Source Code

```
#ifdef __cplusplus
extern "C"{
#endif

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

/* 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;

#ifdef 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("FEEB6AA2-84B1-11d2-BA47-00C04FBFE08B")
ITPCC : public IUnknown
{
public:
    virtual HRESULT __stdcall NewOrder(
        /* [in] */ VARIANT txn_in,
        /* [out] */ VARIANT __RPC_FAR *txn_out) = 0;

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

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

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

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

    virtual HRESULT __stdcall CallSetComplete( void) = 0;

};

#else /* C style interface */

typedef struct ITPCCVtbl
{
    BEGIN_INTERFACE

    HRESULT ( STDMETHODCALLTYPE __RPC_FAR *QueryInterface )(
        ITPCC __RPC_FAR * This,
```

```
        /* [in] */ REFIID riid,
        /* [iid_is][out] */ void __RPC_FAR * __RPC_FAR *ppvObject);

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

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

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

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

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

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

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

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

    END_INTERFACE
} ITPCCVtbl;

interface ITPCC
{
    CONST_VTBL struct ITPCCVtbl __RPC_FAR *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)
```


Appendix A - Application Source Code

```
#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 __stdcall ITPCC_NewOrder_Proxy(
    ITPCC __RPC_FAR * This,
    /* [in] */ VARIANT txn_in,
    /* [out] */ VARIANT __RPC_FAR *txn_out);

void __RPC_STUB ITPCC_NewOrder_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * pRpcChannelBuffer,
    PRPC_MESSAGE pRpcMessage,
    DWORD *pdwStubPhase);

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

void __RPC_STUB ITPCC_Payment_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * pRpcChannelBuffer,
    PRPC_MESSAGE pRpcMessage,
    DWORD *pdwStubPhase);

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

void __RPC_STUB ITPCC_Delivery_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * pRpcChannelBuffer,
    PRPC_MESSAGE pRpcMessage,
    DWORD *pdwStubPhase);

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

void __RPC_STUB ITPCC_StockLevel_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * pRpcChannelBuffer,
    PRPC_MESSAGE pRpcMessage,
    DWORD *pdwStubPhase);

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

void __RPC_STUB ITPCC_OrderStatus_Stub(
    IRpcStubBuffer *This,
    IRpcChannelBuffer * pRpcChannelBuffer,
    PRPC_MESSAGE pRpcMessage,
    DWORD *pdwStubPhase);

HRESULT __stdcall ITPCC_CallSetComplete_Proxy(
    ITPCC __RPC_FAR * 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 __RPC_FAR *,
unsigned long             , VARIANT __RPC_FAR * );
unsigned char             __RPC_USER VARIANT_UserMarshal(  unsigned long __RPC_FAR *,
unsigned char             __RPC_FAR *, VARIANT __RPC_FAR * );
unsigned char             __RPC_USER VARIANT_UserUnmarshal(unsigned long __RPC_FAR *,
unsigned char             __RPC_FAR *, VARIANT __RPC_FAR * );
void                     __RPC_USER VARIANT_UserFree(     unsigned long __RPC_FAR *,
VARIANT __RPC_FAR * );

/* end of Additional Prototypes */

#ifdef __cplusplus
}
#endif

#endif
```

Appendix A - Application Source Code

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 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000 */
/*
 * Compiler settings for .\src\tpcc_com_ps.idl:
 * Oicf (OptLev=i2), Wl, Zp8, env=Win32 (32b run), 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_AXP64)

#ifdef __cplusplus
extern "C"{
#endif

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

#ifdef _MIDL_USE_GUIDDEF_

#ifdef 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_

#ifdef __cplusplus
#endif

#endif

#endif
```

Appendix A - Application Source Code

```
#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,
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_AXP64)*/

#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 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000 */
/*
/* Compiler settings for .\src\tpcc_com_ps.idl:
    Oicf (OptLev=i2), Wl, Zp8, env=Win64 (32b run,appending), ms_ext, c_ext, robust
    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( )

#ifdef defined(_M_IA64) || defined(_M_AXP64)

#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,
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_AXP64)*/
```

tpcc_com_ps/src/tpcc_com_ps_p.c

Appendix A - Application Source Code

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

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

/* File created by MIDL compiler version 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000
*/
/* Compiler settings for .\src\tpcc_com_ps.idl:
  Oicf (OptLev=i2), Wl, Zp8, env=Win32 (32b run), 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_AXP64)
#define USE_STUBLESS_PROXY

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

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

#include "tpcc_com_ps.h"

#define TYPE_FORMAT_STRING_SIZE 997
#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;

extern const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString;
extern const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString;

/* Standard interface: __MIDL_itf_tpcc_com_ps_0000, ver. 0.0,
  GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}} */

/* Object interface: IUnknown, ver. 0.0,
  GUID={0x00000000,0x0000,0x0000,{0xc0,0x00,0x00,0x00,0x00,0x00,0x00,0x46}} */
```

```
/* Object interface: ITPCC, ver. 0.0,
  GUID={0xFEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}} */

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO ITPCC_ServerInfo;

#pragma code_seg(".orpc")
static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
    0,
    34,
    68,
    102,
    136,
    170
};

static const MIDL_SERVER_INFO ITPCC_ServerInfo =
{
    &Object_StubDesc,
    0,
    __MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    0,
    0,
    0
};

static const MIDL_STUBLESS_PROXY_INFO ITPCC_ProxyInfo =
{
    &Object_StubDesc,
    __MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    0,
    0
};

CINTERFACE_PROXY_VTABLE(9) _ITPCCProxyVtbl =
{
    &ITPCC_ProxyInfo,
    &IID_ITPCC,
    IUnknown_QueryInterface_Proxy,
    IUnknown_AddRef_Proxy,
    IUnknown_Release_Proxy ,
    (void *)-1 /* ITPCC::NewOrder */ ,
    (void *)-1 /* ITPCC::Payment */ ,
    (void *)-1 /* ITPCC::Delivery */ ,
    (void *)-1 /* ITPCC::StockLevel */ ,
    (void *)-1 /* ITPCC::OrderStatus */ ,
    (void *)-1 /* ITPCC::CallSetComplete */
};

const CInterfaceStubVtbl _ITPCCStubVtbl =
{
    &IID_ITPCC,
    &ITPCC_ServerInfo,
```

Appendix A - Application Source Code

```
9,
0, /* pure interpreted */
CStdStubBuffer_METHODS
);

extern const USER_MARSHAL_ROUTINE_QUADRUPLE UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE
];

static const MIDL_STUB_DESC Object_StubDesc =
{
0,
NdrOleAllocate,
NdrOleFree,
0,
0,
0,
0,
0,
0,
0,
0,
_MIDL_TypeFormatString.Format,
1, /* -error bounds_check flag */
0x20000, /* Ndr library version */
0,
0x5030118, /* MIDL Version 5.3.280 */
0,
UserMarshalRoutines,
0, /* notify & notify_flag routine table */
0x1, /* MIDL flag */
0, /* Reserved3 */
0, /* Reserved4 */
0, /* Reserved5 */
};

#pragma data_seg(".rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE
] =
{
{
VARIANT_UserSize
,VARIANT_UserMarshal
,VARIANT_UserUnmarshal
,VARIANT_UserFree
}
};

#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_marshall] 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 */

0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, Oi2 */

/* 2 */ NdrFcLong( 0x0 ), /* 0 */
/* 6 */ NdrFcShort( 0x3 ), /* 3 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined(_MIPS_)
/* 8 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS Stack size/offset = 32 */
#endif
#endif
#else
NdrFcShort( 0x20 ), /* PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack size/offset = 40 */
#endif
/* 10 */ NdrFcShort( 0x0 ), /* 0 */
/* 12 */ NdrFcShort( 0x8 ), /* 8 */
/* 14 */ 0x7, /* Oi2 Flags: srv must size, clt must size, has return, */
0x3, /* 3 */

/* Parameter txn_in */

/* 16 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined(_MIPS_)
/* 18 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS Stack size/offset = 8 */
#endif
#endif
#else
NdrFcShort( 0x8 ), /* PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack size/offset = 8 */
#endif
/* 20 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 22 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=16 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined(_MIPS_)
/* 24 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS Stack size/offset = 24 */
#endif
#endif
#else
NdrFcShort( 0x18 ), /* PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack size/offset = 24 */
#endif
#endif
#endif
};
```

Appendix A - Application Source Code

```
/* 26 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
/* Return value */
/* 28 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 30 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x1c ), /* PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack size/offset = 32 */
#endif
/* 32 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure Payment */
/* 34 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, Oi2 */
/* 36 */ NdrFcLong( 0x0 ), /* 0 */
/* 40 */ NdrFcShort( 0x4 ), /* 4 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 42 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x20 ), /* PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack size/offset = 40 */
#endif
/* 44 */ NdrFcShort( 0x0 ), /* 0 */
/* 46 */ NdrFcShort( 0x8 ), /* 8 */
/* 48 */ 0x7, /* Oi2 Flags: srv must size, clt must size, has return, */
0x3, /* 3 */
/* Parameter txn_in */
/* 50 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 52 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack size/offset = 8 */
#endif
/* 54 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
```

```
/* Parameter txn_out */
/* 56 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=16 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 58 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack size/offset = 24 */
#endif
/* 60 */ NdrFcShort( 0x3da ), /* Type Offset=986 */
/* Return value */
/* 62 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 64 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x1c ), /* PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack size/offset = 32 */
#endif
/* 66 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure Delivery */
/* 68 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, Oi2 */
/* 70 */ NdrFcLong( 0x0 ), /* 0 */
/* 74 */ NdrFcShort( 0x5 ), /* 5 */
#ifdef _ALPHA_
#ifdef _PPC_
#if !defined( _MIPS_ )
/* 76 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x20 ), /* PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack size/offset = 40 */
#endif
/* 78 */ NdrFcShort( 0x0 ), /* 0 */
/* 80 */ NdrFcShort( 0x8 ), /* 8 */
/* 82 */ 0x7, /* Oi2 Flags: srv must size, clt must size, has return, */
0x3, /* 3 */
/* Parameter txn_in */
```

Appendix A - Application Source Code

```
/* 84 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 86 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack size/offset = 8 */
#endif
/* 88 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 90 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=16 */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 92 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack size/offset = 24 */
#endif
/* 94 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 96 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 98 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x1c ), /* PPC Stack size/offset = 28 */
#endif
#else
NdrFcShort( 0x20 ), /* Alpha Stack size/offset = 32 */
#endif
/* 100 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure StockLevel */

/* 102 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, Oi2 */
/* 104 */ NdrFcLong( 0x0 ), /* 0 */
/* 108 */ NdrFcShort( 0x6 ), /* 6 */
#ifndef ALPHA_
#ifndef PPC_
```

```
#if !defined( MIPS_ )
/* 110 */ NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
NdrFcShort( 0x20 ), /* MIPS Stack size/offset = 32 */
#endif
#endif
#else
NdrFcShort( 0x20 ), /* PPC Stack size/offset = 32 */
#endif
#else
NdrFcShort( 0x28 ), /* Alpha Stack size/offset = 40 */
#endif
/* 112 */ NdrFcShort( 0x0 ), /* 0 */
/* 114 */ NdrFcShort( 0x8 ), /* 8 */
/* 116 */ 0x7, /* Oi2 Flags: srv must size, clt must size, has return, */
0x3, /* 3 */

/* Parameter txn_in */

/* 118 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 120 */ NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
NdrFcShort( 0x8 ), /* MIPS Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* PPC Stack size/offset = 8 */
#endif
#else
NdrFcShort( 0x8 ), /* Alpha Stack size/offset = 8 */
#endif
/* 122 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Parameter txn_out */

/* 124 */ NdrFcShort( 0x4113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=16 */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 126 */ NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
NdrFcShort( 0x18 ), /* MIPS Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* PPC Stack size/offset = 24 */
#endif
#else
NdrFcShort( 0x18 ), /* Alpha Stack size/offset = 24 */
#endif
/* 128 */ NdrFcShort( 0x3da ), /* Type Offset=986 */

/* Return value */

/* 130 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 132 */ NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
NdrFcShort( 0x1c ), /* MIPS Stack size/offset = 28 */
#endif
#endif
#endif
```

Appendix A - Application Source Code

```
#else
    NdrFcShort( 0x1c ), /* PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x20 ), /* Alpha Stack size/offset = 32 */
#endif
/* 134 */ /* 0x8,
    /* FC_LONG */
    0x0, /* 0 */

    /* Procedure OrderStatus */

/* 136 */ /* 0x33,
    /* FC_AUTO_HANDLE */
    0x6c, /* Old Flags: object, Oi2 */
/* 138 */ /* NdrFcLong( 0x0 ), /* 0 */
/* 142 */ /* NdrFcShort( 0x7 ), /* 7 */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 144 */ /* NdrFcShort( 0x1c ), /* x86 Stack size/offset = 28 */
#else
    NdrFcShort( 0x20 ), /* MIPS Stack size/offset = 32 */
#endif
#endif
#else
    NdrFcShort( 0x20 ), /* PPC Stack size/offset = 32 */
#endif
#else
    NdrFcShort( 0x28 ), /* Alpha Stack size/offset = 40 */
#endif
/* 146 */ /* NdrFcShort( 0x0 ), /* 0 */
/* 148 */ /* NdrFcShort( 0x8 ), /* 8 */
/* 150 */ /* 0x7,
    /* Oi2 Flags: srv must size, clt must size, has return, */
    0x3, /* 3 */

    /* Parameter txn_in */

/* 152 */ /* NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 154 */ /* NdrFcShort( 0x4 ), /* x86 Stack size/offset = 4 */
#else
    NdrFcShort( 0x8 ), /* MIPS Stack size/offset = 8 */
#endif
#endif
#else
    NdrFcShort( 0x8 ), /* PPC Stack size/offset = 8 */
#endif
#else
    NdrFcShort( 0x8 ), /* Alpha Stack size/offset = 8 */
#endif
/* 156 */ /* NdrFcShort( 0x3c8 ),
    /* Type Offset=968 */

    /* Parameter txn_out */

/* 158 */ /* NdrFcShort( 0x4113 ),
    /* Flags: must size, must free, out, simple ref,
    srv alloc size=16 */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 160 */ /* NdrFcShort( 0x14 ), /* x86 Stack size/offset = 20 */
#else
    NdrFcShort( 0x18 ), /* MIPS Stack size/offset = 24 */
#endif
#endif
#else
    NdrFcShort( 0x18 ), /* PPC Stack size/offset = 24 */
#endif
#endif
#endif
```

```

    NdrFcShort( 0x18 ), /* PPC Stack size/offset = 24 */
#endif
#else
    NdrFcShort( 0x18 ), /* Alpha Stack size/offset = 24 */
#endif
/* 162 */ /* NdrFcShort( 0x3da ),
    /* Type Offset=986 */

    /* Return value */

/* 164 */ /* NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef ALPHA_
#ifndef PPC_
#if !defined( MIPS_ )
/* 166 */ /* NdrFcShort( 0x18 ), /* x86 Stack size/offset = 24 */
#else
    NdrFcShort( 0x1c ), /* MIPS Stack size/offset = 28 */
#endif
#endif
#else
    NdrFcShort( 0x1c ), /* PPC Stack size/offset = 28 */
#endif
#else
    NdrFcShort( 0x20 ), /* Alpha Stack size/offset = 32 */
#endif
/* 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 */
#ifndef ALPHA_
/* 178 */ /* NdrFcShort( 0x8 ), /* x86, MIPS, PPC Stack size/offset = 8 */
#else
    NdrFcShort( 0x10 ), /* Alpha Stack size/offset = 16 */
#endif
/* 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, */
#ifndef ALPHA_
/* 188 */ /* NdrFcShort( 0x4 ), /* x86, MIPS, PPC Stack size/offset = 4 */
#else
    NdrFcShort( 0x8 ), /* Alpha Stack size/offset = 8 */
#endif
/* 190 */ /* 0x8,
    /* FC_LONG */
    0x0, /* 0 */

    0x0

    };

static const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString =
{
    0,
    {
        NdrFcShort( 0x0 ), /* 0 */
    }
};

/* 2 */
```


Appendix A - Application Source Code

```
/* 4 */ NdrFcShort( 0x3b0 ), /* FC_UP */
/* 6 */
                                0x12, 0x0, /* Offset= 944 (948) */
                                /* FC_NON_ENCAPSULATED_UNION */
                                0x2b, /* FC_ULONGLONG */
                                0x9, /* Corr desc: FC_USHORT */
/* 8 */ 0x7, /* Offset= 944 (948) */
                                0x0, /* FC */
/* 10 */ NdrFcShort( 0xffff8 ), /* -8 */
/* 12 */ NdrFcShort( 0x2 ), /* Offset= 2 (14) */
/* 14 */ NdrFcShort( 0x10 ), /* 16 */
/* 16 */ NdrFcShort( 0x2b ), /* 43 */
/* 18 */ NdrFcLong( 0x3 ), /* 3 */
/* 22 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 24 */ NdrFcLong( 0x11 ), /* 17 */
/* 28 */ NdrFcShort( 0x8001 ), /* Simple arm type: FC_BYTE */
/* 30 */ NdrFcLong( 0x2 ), /* 2 */
/* 34 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 36 */ NdrFcLong( 0x4 ), /* 4 */
/* 40 */ NdrFcShort( 0x800a ), /* Simple arm type: FC_FLOAT */
/* 42 */ NdrFcLong( 0x5 ), /* 5 */
/* 46 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/* 48 */ NdrFcLong( 0xb ), /* 11 */
/* 52 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 54 */ NdrFcLong( 0xa ), /* 10 */
/* 58 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 60 */ NdrFcLong( 0x6 ), /* 6 */
/* 64 */ NdrFcShort( 0xd6 ), /* Offset= 214 (278) */
/* 66 */ NdrFcLong( 0x7 ), /* 7 */
/* 70 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/* 72 */ NdrFcLong( 0x8 ), /* 8 */
/* 76 */ NdrFcShort( 0xd0 ), /* Offset= 208 (284) */
/* 78 */ NdrFcLong( 0xd ), /* 13 */
/* 82 */ NdrFcShort( 0xe2 ), /* Offset= 226 (308) */
/* 84 */ NdrFcLong( 0x9 ), /* 9 */
/* 88 */ NdrFcShort( 0xee ), /* Offset= 238 (326) */
/* 90 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 94 */ NdrFcShort( 0xfa ), /* Offset= 250 (344) */
/* 96 */ NdrFcLong( 0x24 ), /* 36 */
/* 100 */ NdrFcShort( 0x308 ), /* Offset= 776 (876) */
/* 102 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 106 */ NdrFcShort( 0x302 ), /* Offset= 770 (876) */
/* 108 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 112 */ NdrFcShort( 0x300 ), /* Offset= 768 (880) */
/* 114 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 118 */ NdrFcShort( 0x2fe ), /* Offset= 766 (884) */
/* 120 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 124 */ NdrFcShort( 0x2fc ), /* Offset= 764 (888) */
/* 126 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 130 */ NdrFcShort( 0x2fa ), /* Offset= 762 (892) */
/* 132 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 136 */ NdrFcShort( 0x2f8 ), /* Offset= 760 (896) */
/* 138 */ NdrFcLong( 0x400b ), /* 16395 */
/* 142 */ NdrFcShort( 0x2e6 ), /* Offset= 742 (884) */
/* 144 */ NdrFcLong( 0x400a ), /* 16394 */
/* 148 */ NdrFcShort( 0x2e4 ), /* Offset= 740 (888) */
/* 150 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 154 */ NdrFcShort( 0x2ea ), /* Offset= 746 (900) */
/* 156 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 160 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (896) */
/* 162 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 166 */ NdrFcShort( 0x2e2 ), /* Offset= 738 (904) */
/* 168 */ NdrFcLong( 0x400d ), /* 16397 */
/* 172 */ NdrFcShort( 0x2e0 ), /* Offset= 736 (908) */
```

```
/* 174 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 178 */ NdrFcShort( 0x2de ), /* Offset= 734 (912) */
/* 180 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 184 */ NdrFcShort( 0x2dc ), /* Offset= 732 (916) */
/* 186 */ NdrFcLong( 0x400c ), /* 16396 */
/* 190 */ NdrFcShort( 0x2da ), /* Offset= 730 (920) */
/* 192 */ NdrFcLong( 0x10 ), /* 16 */
/* 196 */ NdrFcShort( 0x8002 ), /* Simple arm type: FC_CHAR */
/* 198 */ NdrFcLong( 0x12 ), /* 18 */
/* 202 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 204 */ NdrFcLong( 0x13 ), /* 19 */
/* 208 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 210 */ NdrFcLong( 0x16 ), /* 22 */
/* 214 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 216 */ NdrFcLong( 0x17 ), /* 23 */
/* 220 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 222 */ NdrFcLong( 0xe ), /* 14 */
/* 226 */ NdrFcShort( 0x2be ), /* Offset= 702 (928) */
/* 228 */ NdrFcLong( 0x400e ), /* 16398 */
/* 232 */ NdrFcShort( 0x2c4 ), /* Offset= 708 (940) */
/* 234 */ NdrFcLong( 0x4010 ), /* 16400 */
/* 238 */ NdrFcShort( 0x2c2 ), /* Offset= 706 (944) */
/* 240 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 244 */ NdrFcShort( 0x280 ), /* Offset= 640 (884) */
/* 246 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 250 */ NdrFcShort( 0x27e ), /* Offset= 638 (888) */
/* 252 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 256 */ NdrFcShort( 0x278 ), /* Offset= 632 (888) */
/* 258 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 262 */ NdrFcShort( 0x272 ), /* Offset= 626 (888) */
/* 264 */ NdrFcLong( 0x0 ), /* 0 */
/* 268 */ NdrFcShort( 0x0 ), /* Offset= 0 (268) */
/* 270 */ NdrFcLong( 0x1 ), /* 1 */
/* 274 */ NdrFcShort( 0x0 ), /* Offset= 0 (274) */
/* 276 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (275) */
/* 278 */
                                0x15, /* FC_STRUCT */
                                0x7, /* 7 */
/* 280 */ NdrFcShort( 0x8 ), /* 8 */
/* 282 */ 0xb, /* FC_HYPER */
                                0x5b, /* FC_END */
/* 284 */
                                0x12, 0x0, /* FC_UP */
/* 286 */ NdrFcShort( 0xc ), /* Offset= 12 (298) */
/* 288 */
                                0x1b, /* FC_CARRAY */
                                0x1, /* 1 */
/* 290 */ NdrFcShort( 0x2 ), /* 2 */
/* 292 */ 0x9, /* Corr desc: FC_ULONGLONG */
                                0x0, /* FC */
/* 294 */ NdrFcShort( 0xffffc ), /* -4 */
/* 296 */ 0x6, /* FC_SHORT */
                                0x5b, /* FC_END */
/* 298 */
                                0x17, /* FC_CSTRUCT */
                                0x3, /* 3 */
/* 300 */ NdrFcShort( 0x8 ), /* 8 */
/* 302 */ NdrFcShort( 0xffffffff2 ), /* Offset= -14 (288) */
/* 304 */ 0x8, /* FC_LONG */
                                0x8, /* FC_LONG */
/* 306 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 308 */
```

Appendix A - Application Source Code

```

                                0x2f,          /* FC_IP */
                                0x5a,          /* FC_CONSTANT_IID */
/* 310 */ NdrFcLong( 0x0 ), /* 0 */
/* 314 */ NdrFcShort( 0x0 ), /* 0 */
/* 316 */ NdrFcShort( 0x0 ), /* 0 */
/* 318 */ 0xc0, /* 192 */
                                0x0,          /* 0 */
/* 320 */ 0x0, /* 0 */
                                0x0,          /* 0 */
/* 322 */ 0x0, /* 0 */
                                0x0,          /* 0 */
/* 324 */ 0x0, /* 0 */
                                0x46,          /* 70 */
/* 326 */
                                0x2f,          /* FC_IP */
                                0x5a,          /* FC_CONSTANT_IID */
/* 328 */ NdrFcLong( 0x20400 ), /* 132096 */
/* 332 */ NdrFcShort( 0x0 ), /* 0 */
/* 334 */ NdrFcShort( 0x0 ), /* 0 */
/* 336 */ 0xc0, /* 192 */
                                0x0,          /* 0 */
/* 338 */ 0x0, /* 0 */
                                0x0,          /* 0 */
/* 340 */ 0x0, /* 0 */
                                0x0,          /* 0 */
/* 342 */ 0x0, /* 0 */
                                0x46,          /* 70 */
/* 344 */
                                0x12, 0x10,      /* FC_UP [pointer_deref] */
/* 346 */ NdrFcShort( 0x2 ), /* Offset= 2 (348) */
/* 348 */
                                0x12, 0x0,      /* FC_UP */
/* 350 */ NdrFcShort( 0x1fc ), /* Offset= 508 (858) */
/* 352 */
                                0x2a,          /* FC_ENCAPSULATED_UNION */
                                0x49,          /* 73 */
/* 354 */ NdrFcShort( 0x18 ), /* 24 */
/* 356 */ NdrFcShort( 0xa ), /* 10 */
/* 358 */ NdrFcLong( 0x8 ), /* 8 */
/* 362 */ NdrFcShort( 0x58 ), /* Offset= 88 (450) */
/* 364 */ NdrFcLong( 0xd ), /* 13 */
/* 368 */ NdrFcShort( 0x78 ), /* Offset= 120 (488) */
/* 370 */ NdrFcLong( 0x9 ), /* 9 */
/* 374 */ NdrFcShort( 0x94 ), /* Offset= 148 (522) */
/* 376 */ NdrFcLong( 0xc ), /* 12 */
/* 380 */ NdrFcShort( 0xbc ), /* Offset= 188 (568) */
/* 382 */ NdrFcLong( 0x24 ), /* 36 */
/* 386 */ NdrFcShort( 0x114 ), /* Offset= 276 (662) */
/* 388 */ NdrFcLong( 0x800d ), /* 32781 */
/* 392 */ NdrFcShort( 0x130 ), /* Offset= 304 (696) */
/* 394 */ NdrFcLong( 0x10 ), /* 16 */
/* 398 */ NdrFcShort( 0x148 ), /* Offset= 328 (726) */
/* 400 */ NdrFcLong( 0x2 ), /* 2 */
/* 404 */ NdrFcShort( 0x160 ), /* Offset= 352 (756) */
/* 406 */ NdrFcLong( 0x3 ), /* 3 */
/* 410 */ NdrFcShort( 0x178 ), /* Offset= 376 (786) */
/* 412 */ NdrFcLong( 0x14 ), /* 20 */
/* 416 */ NdrFcShort( 0x190 ), /* Offset= 400 (816) */
/* 418 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (417) */
/* 420 */
                                0x1b,          /* FC_CARRAY */
                                0x3,          /* 3 */
/* 422 */ NdrFcShort( 0x4 ), /* 4 */

                                0x19,          /* Corr desc: field pointer, FC_ULONG */
/* 424 */ 0x0, /* */
/* 426 */ NdrFcShort( 0x0 ), /* 0 */
/* 428 */
                                0x4b,          /* FC_PP */
                                0x5c,          /* FC_PAD */
/* 430 */
                                0x48,          /* FC_VARIABLE_REPEAT */
                                0x49,          /* FC_FIXED_OFFSET */
/* 432 */ NdrFcShort( 0x4 ), /* 4 */
/* 434 */ NdrFcShort( 0x0 ), /* 0 */
/* 436 */ NdrFcShort( 0x1 ), /* 1 */
/* 438 */ NdrFcShort( 0x0 ), /* 0 */
/* 440 */ NdrFcShort( 0x0 ), /* 0 */
/* 442 */ 0x12, 0x0, /* FC_UP */
/* 444 */ NdrFcShort( 0xfffff6e ), /* Offset= -146 (298) */
/* 446 */
                                0x5b,          /* FC_END */
/* 448 */ 0x5c, /* FC_LONG */
                                0x5b,          /* FC_PAD */
/* 450 */
                                0x16,          /* FC_PSTRUCT */
                                0x3,          /* 3 */
/* 452 */ NdrFcShort( 0x8 ), /* 8 */
/* 454 */
                                0x4b,          /* FC_PP */
                                0x5c,          /* FC_PAD */
/* 456 */
                                0x46,          /* FC_NO_REPEAT */
                                0x5c,          /* FC_PAD */
/* 458 */ NdrFcShort( 0x4 ), /* 4 */
/* 460 */ NdrFcShort( 0x4 ), /* 4 */
/* 462 */ 0x11, 0x0, /* FC_RP */
/* 464 */ NdrFcShort( 0xffffffff4 ), /* Offset= -44 (420) */
/* 466 */
                                0x5b,          /* FC_END */
/* 468 */ 0x8, /* FC_LONG */
                                0x5b,          /* FC_PAD */
/* 470 */
                                0x21,          /* FC_BOGUS_ARRAY */
                                0x3,          /* 3 */
/* 472 */ NdrFcShort( 0x0 ), /* 0 */
/* 474 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0,          /* */
/* 476 */ NdrFcShort( 0x0 ), /* 0 */
/* 478 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 482 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0,          /* 0 */
/* 484 */ NdrFcShort( 0xffffffff50 ), /* Offset= -176 (308) */
/* 486 */ 0x5c, /* FC_PAD */
                                0x5b,          /* FC_END */
/* 488 */
                                0x1a,          /* FC_BOGUS_STRUCT */
                                0x3,          /* 3 */
/* 490 */ NdrFcShort( 0x8 ), /* 8 */
/* 492 */ NdrFcShort( 0x0 ), /* 0 */
/* 494 */ NdrFcShort( 0x6 ), /* Offset= 6 (500) */
/* 496 */ 0x8, /* FC_LONG */
                                0x36,          /* FC_POINTER */
```

Appendix A - Application Source Code

```
/* 498 */ 0x5c,          /* FC_PAD */
/* 500 */                                0x5b,          /* FC_END */
/* 502 */ NdrFcShort( 0xffffffe0 ), /* Offset= -32 (470) */
/* 504 */
                                0x21,          /* FC_BOGUS_ARRAY */
                                0x3,          /* 3 */
/* 506 */ NdrFcShort( 0x0 ), /* 0 */
/* 508 */ 0x19,          /* Corr desc: field pointer, FC_ULONG */
                                0x0,          /* */
/* 510 */ NdrFcShort( 0x0 ), /* 0 */
/* 512 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 516 */ 0x4c,          /* FC_EMBEDDED_COMPLEX */
                                0x0,          /* 0 */
/* 518 */ NdrFcShort( 0xfffffff40 ), /* Offset= -192 (326) */
/* 520 */ 0x5c,          /* FC_PAD */
                                0x5b,          /* FC_END */
/* 522 */
                                0x1a,          /* FC_BOGUS_STRUCT */
                                0x3,          /* 3 */
/* 524 */ NdrFcShort( 0x8 ), /* 8 */
/* 526 */ NdrFcShort( 0x0 ), /* 0 */
/* 528 */ NdrFcShort( 0x6 ), /* Offset= 6 (534) */
/* 530 */ 0x8,          /* FC_LONG */
                                0x36,          /* FC_POINTER */
/* 532 */ 0x5c,          /* FC_PAD */
                                0x5b,          /* FC_END */
/* 534 */
                                0x11, 0x0,      /* FC_RP */
/* 536 */ NdrFcShort( 0xffffffe0 ), /* Offset= -32 (504) */
/* 538 */
                                0x1b,          /* FC_CARRAY */
                                0x3,          /* 3 */
/* 540 */ NdrFcShort( 0x4 ), /* 4 */
/* 542 */ 0x19,          /* Corr desc: field pointer, FC_ULONG */
                                0x0,          /* */
/* 544 */ NdrFcShort( 0x0 ), /* 0 */
/* 546 */
                                0x4b,          /* FC_PP */
                                0x5c,          /* FC_PAD */
/* 548 */
                                0x48,          /* FC_VARIABLE_REPEAT */
                                0x49,          /* FC_FIXED_OFFSET */
/* 550 */ NdrFcShort( 0x4 ), /* 4 */
/* 552 */ NdrFcShort( 0x0 ), /* 0 */
/* 554 */ NdrFcShort( 0x1 ), /* 1 */
/* 556 */ NdrFcShort( 0x0 ), /* 0 */
/* 558 */ NdrFcShort( 0x0 ), /* 0 */
/* 560 */ 0x12, 0x0,      /* FC_UP */
/* 562 */ NdrFcShort( 0x182 ), /* Offset= 386 (948) */
/* 564 */
                                0x5b,          /* FC_END */
/* 566 */ 0x5c,          /* FC_LONG */
                                0x5b,          /* FC_END */
/* 568 */
                                0x1a,          /* FC_BOGUS_STRUCT */
                                0x3,          /* 3 */
/* 570 */ NdrFcShort( 0x8 ), /* 8 */
/* 572 */ NdrFcShort( 0x0 ), /* 0 */
/* 574 */ NdrFcShort( 0x6 ), /* Offset= 6 (580) */

/* 576 */ 0x8,          /* FC_LONG */
                                0x36,          /* FC_POINTER */
/* 578 */ 0x5c,          /* FC_PAD */
                                0x5b,          /* FC_END */
/* 580 */
                                0x11, 0x0,      /* FC_RP */
/* 582 */ NdrFcShort( 0xffffffd4 ), /* Offset= -44 (538) */
/* 584 */
                                0x2f,          /* FC_IP */
                                0x5a,          /* FC_CONSTANT_IID */
/* 586 */ NdrFcLong( 0x2f ), /* 47 */
/* 590 */ NdrFcShort( 0x0 ), /* 0 */
/* 592 */ NdrFcShort( 0x0 ), /* 0 */
/* 594 */ 0xc0,          /* 192 */
                                0x0,          /* 0 */
/* 596 */ 0x0,          /* 0 */
                                0x0,          /* 0 */
/* 598 */ 0x0,          /* 0 */
                                0x0,          /* 0 */
/* 600 */ 0x0,          /* 0 */
                                0x46,          /* 70 */
/* 602 */
                                0x1b,          /* FC_CARRAY */
                                0x0,          /* 0 */
/* 604 */ NdrFcShort( 0x1 ), /* 1 */
/* 606 */ 0x19,          /* Corr desc: field pointer, FC_ULONG */
                                0x0,          /* */
/* 608 */ NdrFcShort( 0x4 ), /* 4 */
/* 610 */ 0x1,          /* FC_BYTE */
                                0x5b,          /* FC_END */
/* 612 */
                                0x1a,          /* FC_BOGUS_STRUCT */
                                0x3,          /* 3 */
/* 614 */ NdrFcShort( 0x10 ), /* 16 */
/* 616 */ NdrFcShort( 0x0 ), /* 0 */
/* 618 */ NdrFcShort( 0xa ), /* Offset= 10 (628) */
/* 620 */ 0x8,          /* FC_LONG */
                                0x8,          /* FC_LONG */
                                0x0,          /* FC_EMBEDDED_COMPLEX */
/* 622 */ 0x4c,          /* 0 */
/* 624 */ NdrFcShort( 0xffffffd8 ), /* Offset= -40 (584) */
/* 626 */ 0x36,          /* FC_POINTER */
                                0x5b,          /* FC_END */
/* 628 */
                                0x12, 0x0,      /* FC_UP */
/* 630 */ NdrFcShort( 0xffffffe4 ), /* Offset= -28 (602) */
/* 632 */
                                0x1b,          /* FC_CARRAY */
                                0x3,          /* 3 */
/* 634 */ NdrFcShort( 0x4 ), /* 4 */
/* 636 */ 0x19,          /* Corr desc: field pointer, FC_ULONG */
                                0x0,          /* */
/* 638 */ NdrFcShort( 0x0 ), /* 0 */
/* 640 */
                                0x4b,          /* FC_PP */
                                0x5c,          /* FC_PAD */
/* 642 */
                                0x48,          /* FC_VARIABLE_REPEAT */
                                0x49,          /* FC_FIXED_OFFSET */
/* 644 */ NdrFcShort( 0x4 ), /* 4 */
/* 646 */ NdrFcShort( 0x0 ), /* 0 */
/* 648 */ NdrFcShort( 0x1 ), /* 1 */
/* 650 */ NdrFcShort( 0x0 ), /* 0 */
```

Appendix A - Application Source Code

```
/* 652 */ NdrFcShort( 0x0 ), /* 0 */
/* 654 */ 0x12, 0x0, /* FC_UP */
/* 656 */ NdrFcShort( 0xffffffffd4 ), /* Offset= -44 (612) */
/* 658 */
                                0x5b, /* FC_END */
                                0x8, /* FC_LONG */
/* 660 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 662 */
                                0x1a, /* FC_BOGUS_STRUCT */
                                0x3, /* 3 */
/* 664 */ NdrFcShort( 0x8 ), /* 8 */
/* 666 */ NdrFcShort( 0x0 ), /* 0 */
/* 668 */ NdrFcShort( 0x6 ), /* Offset= 6 (674) */
/* 670 */ 0x8, /* FC_LONG */
                                0x36, /* FC_POINTER */
/* 672 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 674 */
                                0x11, 0x0, /* FC_RP */
/* 676 */ NdrFcShort( 0xffffffffd4 ), /* Offset= -44 (632) */
/* 678 */
                                0x1d, /* FC_SMFARRAY */
                                0x0, /* 0 */
/* 680 */ NdrFcShort( 0x8 ), /* 8 */
/* 682 */ 0x2, /* FC_CHAR */
                                0x5b, /* FC_END */
/* 684 */
                                0x15, /* FC_STRUCT */
                                0x3, /* 3 */
/* 686 */ NdrFcShort( 0x10 ), /* 16 */
/* 688 */ 0x8, /* FC_LONG */
                                0x6, /* FC_SHORT */
/* 690 */ 0x6, /* FC_SHORT */
                                0x4c, /* FC_EMBEDDED_COMPLEX */
/* 692 */ 0x0, /* 0 */
                                NdrFcShort( 0xfffffffff1 ), /* Offset= -15 (678) */
                                0x5b, /* FC_END */
/* 696 */
                                0x1a, /* FC_BOGUS_STRUCT */
                                0x3, /* 3 */
/* 698 */ NdrFcShort( 0x18 ), /* 24 */
/* 700 */ NdrFcShort( 0x0 ), /* 0 */
/* 702 */ NdrFcShort( 0xa ), /* Offset= 10 (712) */
/* 704 */ 0x8, /* FC_LONG */
                                0x36, /* FC_POINTER */
/* 706 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0, /* 0 */
/* 708 */ NdrFcShort( 0xffffffffe8 ), /* Offset= -24 (684) */
/* 710 */ 0x5c, /* FC_PAD */
                                0x5b, /* FC_END */
/* 712 */
                                0x11, 0x0, /* FC_RP */
/* 714 */ NdrFcShort( 0xffffffff0c ), /* Offset= -244 (470) */
/* 716 */
                                0x1b, /* FC_CARRAY */
                                0x0, /* 0 */
/* 718 */ NdrFcShort( 0x1 ), /* 1 */
/* 720 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 722 */ NdrFcShort( 0x0 ), /* 0 */
/* 724 */ 0x1, /* FC_BYTE */
                                0x5b, /* FC_END */
                                0x16, /* FC_PSTRUCT */
                                0x3, /* 3 */
/* 726 */
                                0x16, /* FC_PSTRUCT */
                                0x3, /* 3 */
/* 728 */ NdrFcShort( 0x8 ), /* 8 */
/* 730 */
                                0x4b, /* FC_PP */
                                0x5c, /* FC_PAD */
/* 732 */
                                0x46, /* FC_NO_REPEAT */
                                0x5c, /* FC_PAD */
/* 734 */ NdrFcShort( 0x4 ), /* 4 */
/* 736 */ NdrFcShort( 0x4 ), /* 4 */
/* 738 */ 0x12, 0x0, /* FC_UP */
/* 740 */ NdrFcShort( 0xffffffffe8 ), /* Offset= -24 (716) */
/* 742 */
                                0x5b, /* FC_END */
                                0x8, /* FC_LONG */
/* 744 */ 0x8, /* FC_LONG */
                                0x5b, /* FC_END */
/* 746 */
                                0x1b, /* FC_CARRAY */
                                0x1, /* 1 */
/* 748 */ NdrFcShort( 0x2 ), /* 2 */
/* 750 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 752 */ NdrFcShort( 0x0 ), /* 0 */
/* 754 */ 0x6, /* FC_SHORT */
                                0x5b, /* FC_END */
/* 756 */
                                0x16, /* FC_PSTRUCT */
                                0x3, /* 3 */
/* 758 */ NdrFcShort( 0x8 ), /* 8 */
/* 760 */
                                0x4b, /* FC_PP */
                                0x5c, /* FC_PAD */
/* 762 */
                                0x46, /* FC_NO_REPEAT */
                                0x5c, /* FC_PAD */
/* 764 */ NdrFcShort( 0x4 ), /* 4 */
/* 766 */ NdrFcShort( 0x4 ), /* 4 */
/* 768 */ 0x12, 0x0, /* FC_UP */
/* 770 */ NdrFcShort( 0xffffffffe8 ), /* Offset= -24 (746) */
/* 772 */
                                0x5b, /* FC_END */
                                0x8, /* FC_LONG */
/* 774 */ 0x8, /* FC_LONG */
                                0x5b, /* FC_END */
/* 776 */
                                0x1b, /* FC_CARRAY */
                                0x3, /* 3 */
/* 778 */ NdrFcShort( 0x4 ), /* 4 */
/* 780 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0, /* */
/* 782 */ NdrFcShort( 0x0 ), /* 0 */
/* 784 */ 0x8, /* FC_LONG */
                                0x5b, /* FC_END */
/* 786 */
                                0x16, /* FC_PSTRUCT */
                                0x3, /* 3 */
/* 788 */ NdrFcShort( 0x8 ), /* 8 */
```

Appendix A - Application Source Code

```
/* 790 */
                                0x4b,          /* FC_PP */
                                0x5c,          /* FC_PAD */
/* 792 */
                                0x46,          /* FC_NO_REPEAT */
                                0x5c,          /* FC_PAD */
/* 794 */ NdrFcShort( 0x4 ), /* 4 */
/* 796 */ NdrFcShort( 0x4 ), /* 4 */
/* 798 */ 0x12, 0x0, /* FC_UP */
/* 800 */ NdrFcShort( 0xffffffe8 ), /* Offset= -24 (776) */
/* 802 */
                                0x5b,          /* FC_END */
/* 804 */ 0x8,
                                0x8,          /* FC_LONG */
                                /* FC_LONG */
                                0x5b,          /* FC_END */
/* 806 */
                                0x1b,          /* FC_CARRAY */
                                0x7,          /* 7 */
/* 808 */ NdrFcShort( 0x8 ), /* 8 */
/* 810 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
                                0x0,          /* */
/* 812 */ NdrFcShort( 0x0 ), /* 0 */
/* 814 */ 0xb, /* FC_HYPER */
                                0x5b,          /* FC_END */
/* 816 */
                                0x16,          /* FC_PSTRUCT */
                                0x3,          /* 3 */
/* 818 */ NdrFcShort( 0x8 ), /* 8 */
/* 820 */
                                0x4b,          /* FC_PP */
                                0x5c,          /* FC_PAD */
/* 822 */
                                0x46,          /* FC_NO_REPEAT */
                                0x5c,          /* FC_PAD */
/* 824 */ NdrFcShort( 0x4 ), /* 4 */
/* 826 */ NdrFcShort( 0x4 ), /* 4 */
/* 828 */ 0x12, 0x0, /* FC_UP */
/* 830 */ NdrFcShort( 0xffffffe8 ), /* Offset= -24 (806) */
/* 832 */
                                0x5b,          /* FC_END */
/* 834 */ 0x8,
                                0x8,          /* FC_LONG */
                                /* FC_LONG */
                                0x5b,          /* FC_END */
/* 836 */
                                0x15,          /* FC_STRUCT */
                                0x3,          /* 3 */
/* 838 */ NdrFcShort( 0x8 ), /* 8 */
/* 840 */ 0x8, /* FC_LONG */
                                /* FC_LONG */
/* 842 */ 0x5c, /* FC_PAD */
                                0x5b,          /* FC_END */
/* 844 */
                                0x1b,          /* FC_CARRAY */
                                0x3,          /* 3 */
/* 846 */ NdrFcShort( 0x8 ), /* 8 */
/* 848 */ 0x7, /* Corr desc: FC_USHORT */
                                0x0,          /* */
/* 850 */ NdrFcShort( 0xfffd8 ), /* -40 */
/* 852 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
                                0x0,          /* 0 */
/* 854 */ NdrFcShort( 0xfffffee ), /* Offset= -18 (836) */

/* 856 */ 0x5c, /* FC_PAD */
                                0x5b,          /* FC_END */
/* 858 */
                                0x1a,          /* FC_BOGUS_STRUCT */
                                0x3,          /* 3 */
/* 860 */ NdrFcShort( 0x28 ), /* 40 */
/* 862 */ NdrFcShort( 0xfffffee ), /* Offset= -18 (844) */
/* 864 */ NdrFcShort( 0x0 ), /* Offset= 0 (864) */
/* 866 */ 0x6, /* FC_SHORT */
                                0x6,          /* FC_SHORT */
/* 868 */ 0x38, /* FC_ALIGNM4 */
                                0x8,          /* FC_LONG */
/* 870 */ 0x8, /* FC_LONG */
                                0x4c,          /* FC_EMBEDDED_COMPLEX */
/* 872 */ 0x0, /* 0 */
                                NdrFcShort( 0xfffffd7 ), /* Offset= -521 (352) */
                                0x5b,          /* FC_END */
/* 876 */
                                0x12, 0x0, /* FC_UP */
/* 878 */ NdrFcShort( 0xfffffef6 ), /* Offset= -266 (612) */
/* 880 */
                                0x12, 0x8, /* FC_UP [simple_pointer] */
/* 882 */ 0x1, /* FC_BYTE */
                                0x5c,          /* FC_PAD */
/* 884 */
                                0x12, 0x8, /* FC_UP [simple_pointer] */
/* 886 */ 0x6, /* FC_SHORT */
                                0x5c,          /* FC_PAD */
/* 888 */
                                0x12, 0x8, /* FC_UP [simple_pointer] */
/* 890 */ 0x8, /* FC_LONG */
                                0x5c,          /* FC_PAD */
/* 892 */
                                0x12, 0x8, /* FC_UP [simple_pointer] */
/* 894 */ 0xa, /* FC_FLOAT */
                                0x5c,          /* FC_PAD */
/* 896 */
                                0x12, 0x8, /* FC_UP [simple_pointer] */
/* 898 */ 0xc, /* FC_DOUBLE */
                                0x5c,          /* FC_PAD */
/* 900 */
                                0x12, 0x0, /* FC_UP */
/* 902 */ NdrFcShort( 0xfffffd90 ), /* Offset= -624 (278) */
/* 904 */
                                0x12, 0x10, /* FC_UP [pointer_deref] */
/* 906 */ NdrFcShort( 0xfffffd92 ), /* Offset= -622 (284) */
/* 908 */
                                0x12, 0x10, /* FC_UP [pointer_deref] */
/* 910 */ NdrFcShort( 0xffffda6 ), /* Offset= -602 (308) */
/* 912 */
                                0x12, 0x10, /* FC_UP [pointer_deref] */
/* 914 */ NdrFcShort( 0xfffffdb4 ), /* Offset= -588 (326) */
/* 916 */
                                0x12, 0x10, /* FC_UP [pointer_deref] */
/* 918 */ NdrFcShort( 0xfffffdc2 ), /* Offset= -574 (344) */
/* 920 */
                                0x12, 0x10, /* FC_UP [pointer_deref] */
/* 922 */ NdrFcShort( 0x2 ), /* Offset= 2 (924) */
/* 924 */
                                0x12, 0x0, /* FC_UP */
/* 926 */ NdrFcShort( 0x16 ), /* Offset= 22 (948) */
/* 928 */
                                0x15,          /* FC_STRUCT */
```

Appendix A - Application Source Code

```
/* 930 */ NdrFcShort( 0x10 ), /* 16 */
/* 932 */ 0x6, /* FC_SHORT */
/* 934 */ 0x1, /* FC_BYTE */
/* 936 */ 0x8, /* FC_LONG */
/* 938 */ 0xb, /* FC_HYPER */
/* 940 */
/* 942 */ NdrFcShort( 0xffffffff2 ), /* Offset= -14 (928) */
/* 944 */
/* 946 */ 0x2, /* FC_CHAR */
/* 948 */
/* 950 */ NdrFcShort( 0x20 ), /* 32 */
/* 952 */ NdrFcShort( 0x0 ), /* 0 */
/* 954 */ NdrFcShort( 0x0 ), /* Offset= 0 (954) */
/* 956 */ 0x8, /* FC_LONG */
/* 958 */ 0x6, /* FC_SHORT */
/* 960 */ 0x6, /* FC_SHORT */
/* 962 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 964 */ NdrFcShort( 0xfffffc42 ), /* Offset= -958 (6) */
/* 966 */ 0x5c, /* FC_PAD */
/* 968 */ 0xb4, /* FC_USER_MARSHAL */
/* 970 */ NdrFcShort( 0x0 ), /* 0 */
/* 972 */ NdrFcShort( 0x10 ), /* 16 */
/* 974 */ NdrFcShort( 0x0 ), /* 0 */
/* 976 */ NdrFcShort( 0xfffffc32 ), /* Offset= -974 (2) */
/* 978 */
/* 980 */ NdrFcShort( 0x6 ), /* Offset= 6 (986) */
/* 982 */
/* 984 */ NdrFcShort( 0xfffffcdc ), /* Offset= -36 (948) */
/* 986 */ 0xb4, /* FC_USER_MARSHAL */
/* 988 */ NdrFcShort( 0x0 ), /* 0 */
/* 990 */ NdrFcShort( 0x10 ), /* 16 */
/* 992 */ NdrFcShort( 0x0 ), /* 0 */
/* 994 */ NdrFcShort( 0xfffffff4 ), /* Offset= -12 (982) */
0x0
}
};
const CInterfaceProxyVtbl * _tpcc_com_ps_ProxyVtblList[] =
{
( CInterfaceProxyVtbl *) &_ITPCCProxyVtbl,
0
};
```

```
const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[] =
{
( CInterfaceStubVtbl *) &_ITPCCStubVtbl,
0
};
PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[] =
{
"ITPCC",
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_AXP64) */
#pragma warning( disable: 4049 ) /* more than 64k source lines */
/* this ALWAYS GENERATED file contains the proxy stub code */
/* File created by MIDL compiler version 5.03.0280 */
/* at Sat Apr 08 16:40:10 2000 */
/* Compiler settings for .\src\tpcc_com_ps.idl:
Oicf (OptLev=i2), Wl, Zp8, env=Win64 (32b run,appending), ms_ext, c_ext, robust
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( )
```

Appendix A - Application Source Code

```
#if defined(_M_IA64) || defined(_M_XPP64)
#define USE_STUBLESS_PROXY

/* verify that the <rpcproxy.h> version is high enough to compile this file*/
#ifndef REDO_RPCPROXY_H_VERSION
#define REQUIRED_RPCPROXY_H_VERSION 475
#endif

#include "rpcproxy.h"
#ifndef 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 979
#define PROC_FORMAT_STRING_SIZE 253
#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;

extern const MIDL_TYPE_FORMAT_STRING __MIDL_TypeFormatString;
extern const MIDL_PROC_FORMAT_STRING __MIDL_ProcFormatString;

/* Standard interface: __MIDL_itf_tpcc_com_ps_0000, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00}} */

/* Object interface: IUnknown, ver. 0.0,
GUID={0x00000000,0x0000,0x0000,{0xC0,0x00,0x00,0x00,0x00,0x00,0x00,0x46}} */

/* Object interface: ITPCC, ver. 0.0,
GUID={0xFEEE6AA2,0x84B1,0x11d2,{0xBA,0x47,0x00,0xC0,0x4F,0xBF,0xE0,0x8B}} */

extern const MIDL_STUB_DESC Object_StubDesc;

extern const MIDL_SERVER_INFO ITPCC_ServerInfo;

#pragma code_seg(".orpc")
static const unsigned short ITPCC_FormatStringOffsetTable[] =
{
    0,
    44,
    88,
```

```
132,
176,
220
};

static const MIDL_SERVER_INFO ITPCC_ServerInfo =
{
    &Object_StubDesc,
    0,
    __MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    0,
    0,
    0
};

static const MIDL_STUBLESS_PROXY_INFO ITPCC_ProxyInfo =
{
    &Object_StubDesc,
    __MIDL_ProcFormatString.Format,
    &ITPCC_FormatStringOffsetTable[-3],
    0,
    0,
    0
};

CINTERFACE_PROXY_VTABLE(9) _ITPCCProxyVtbl =
{
    &ITPCC_ProxyInfo,
    &IID_ITPCC,
    IUnknown_QueryInterface_Proxy,
    IUnknown_AddRef_Proxy,
    IUnknown_Release_Proxy ,
    (void *)-1 /* ITPCC::NewOrder */ ,
    (void *)-1 /* ITPCC::Payment */ ,
    (void *)-1 /* ITPCC::Delivery */ ,
    (void *)-1 /* ITPCC::StockLevel */ ,
    (void *)-1 /* ITPCC::OrderStatus */ ,
    (void *)-1 /* ITPCC::CallSetComplete */
};

const CInterfaceStubVtbl _ITPCCStubVtbl =
{
    &IID_ITPCC,
    &ITPCC_ServerInfo,
    9,
    0, /* pure interpreted */
    CStdStubBuffer_METHODS
};

extern const USER_MARSHAL_ROUTINE_QUADRUPLE UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE
];

static const MIDL_STUB_DESC Object_StubDesc =
{
    0,
    NdrOleAllocate,
    NdrOleFree,
    0,
    0,
    0,
    0,
```

Appendix A - Application Source Code

```
0,
    __MIDL_TypeFormatString.Format,
    1, /* -error bounds_check flag */
    0x50002, /* Ndr library version */
    0,
    0x5030118, /* MIDL Version 5.3.280 */
    0,
    UserMarshalRoutines,
    0, /* notify & notify_flag routine table */
    0x1, /* MIDL flag */
    0, /* Reserved3 */
    0, /* Reserved4 */
    0 /* Reserved5 */
};

#pragma data_seg(".rdata")

static const USER_MARSHAL_ROUTINE_QUADRUPLE UserMarshalRoutines[ WIRE_MARSHAL_TABLE_SIZE ] =
{
    {
        VARIANT_UserSize
        ,VARIANT_UserMarshal
        ,VARIANT_UserUnmarshal
        ,VARIANT_UserFree
    }
};

#if !defined(__RPC_WIN64__)
#error Invalid build platform for this stub.
#endif

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

        0x33, /* FC_AUTO_HANDLE */
        0x6c, /* Old Flags: object, Oi2 */
        /* 2 */ NdrFcLong( 0x0 ), /* 0 */
        /* 6 */ NdrFcShort( 0x3 ), /* 3 */
#ifdef ALPHA
        /* 8 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
        NdrFcShort( 0x30 ), /* xpp64 Stack size/offset = 48 */
#endif
        /* 10 */ NdrFcShort( 0x0 ), /* 0 */
        /* 12 */ NdrFcShort( 0x8 ), /* 8 */
        /* 14 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has return,
has ext, */
        0x3, /* 3 */
        /* 16 */ 0xa, /* 10 */
        0x7, /* Ext Flags: new corr desc, clt corr
check, srv corr check, */
        /* 18 */ NdrFcShort( 0x20 ), /* 32 */
        /* 20 */ NdrFcShort( 0x20 ), /* 32 */
        /* 22 */ NdrFcShort( 0x0 ), /* 0 */
        /* 24 */ NdrFcShort( 0x0 ), /* 0 */
    }
};
```

```
/* Parameter txn_in */

/* 26 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef ALPHA
/* 28 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
        NdrFcShort( 0x8 ), /* xpp64 Stack size/offset = 8 */
#endif
/* 30 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */

/* Parameter txn_out */

/* 32 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=24 */
#ifdef ALPHA
/* 34 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
        NdrFcShort( 0x20 ), /* xpp64 Stack size/offset = 32 */
#endif
/* 36 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Return value */

/* 38 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef ALPHA
/* 40 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
        NdrFcShort( 0x28 ), /* xpp64 Stack size/offset = 40 */
#endif
/* 42 */ 0x8, /* FC_LONG */
0x0, /* 0 */

/* Procedure Payment */

/* 44 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, Oi2 */
/* 46 */ NdrFcLong( 0x0 ), /* 0 */
/* 50 */ NdrFcShort( 0x4 ), /* 4 */
#ifdef ALPHA
/* 52 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
        NdrFcShort( 0x30 ), /* xpp64 Stack size/offset = 48 */
#endif
/* 54 */ NdrFcShort( 0x0 ), /* 0 */
/* 56 */ NdrFcShort( 0x8 ), /* 8 */
/* 58 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has return,
has ext, */
0x3, /* 3 */
/* 60 */ 0xa, /* 10 */
0x7, /* Ext Flags: new corr desc, clt corr
check, srv corr check, */
/* 62 */ NdrFcShort( 0x20 ), /* 32 */
/* 64 */ NdrFcShort( 0x20 ), /* 32 */
/* 66 */ NdrFcShort( 0x0 ), /* 0 */
/* 68 */ NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 70 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef ALPHA
/* 72 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
```


Appendix A - Application Source Code

```

                                NdrFcShort( 0x8 ), /* axp64 Stack size/offset = 8 */
#endif
/* 74 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */

/* Parameter txn_out */

/* 76 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=24 */
#ifndef ALPHA
/* 78 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
                                NdrFcShort( 0x20 ), /* axp64 Stack size/offset = 32 */
#endif
/* 80 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Return value */

/* 82 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef ALPHA
/* 84 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
                                NdrFcShort( 0x28 ), /* axp64 Stack size/offset = 40 */
#endif
/* 86 */ 0x8, /* FC_LONG */ /* 0 */

/* Procedure Delivery */

/* 88 */ 0x33, /* FC_AUTO_HANDLE */
                                0x6c, /* Old Flags: object, Oi2 */
/* 90 */ NdrFcLong( 0x0 ), /* 0 */
/* 94 */ NdrFcShort( 0x5 ), /* 5 */
#ifndef ALPHA
/* 96 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
                                NdrFcShort( 0x30 ), /* axp64 Stack size/offset = 48 */
#endif
/* 98 */ NdrFcShort( 0x0 ), /* 0 */
/* 100 */ NdrFcShort( 0x8 ), /* 8 */
/* 102 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has return,
has ext, */
/* 104 */ 0xa, 0x3, /* 3 */
                                0x7, /* 10 */ /* Ext Flags: new corr desc, clt corr
check, srv corr check, */
/* 106 */ NdrFcShort( 0x20 ), /* 32 */
/* 108 */ NdrFcShort( 0x20 ), /* 32 */
/* 110 */ NdrFcShort( 0x0 ), /* 0 */
/* 112 */ NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 114 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifndef ALPHA
/* 116 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
                                NdrFcShort( 0x8 ), /* axp64 Stack size/offset = 8 */
#endif
/* 118 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */

/* Parameter txn_out */
```

```

/* 120 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=24 */
#ifndef ALPHA
/* 122 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
                                NdrFcShort( 0x20 ), /* axp64 Stack size/offset = 32 */
#endif
/* 124 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */

/* Return value */

/* 126 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifndef ALPHA
/* 128 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
                                NdrFcShort( 0x28 ), /* axp64 Stack size/offset = 40 */
#endif
/* 130 */ 0x8, /* FC_LONG */ /* 0 */

/* Procedure StockLevel */

/* 132 */ 0x33, /* FC_AUTO_HANDLE */
                                0x6c, /* Old Flags: object, Oi2 */
/* 134 */ NdrFcLong( 0x0 ), /* 0 */
/* 138 */ NdrFcShort( 0x6 ), /* 6 */
#ifndef ALPHA
/* 140 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
                                NdrFcShort( 0x30 ), /* axp64 Stack size/offset = 48 */
#endif
/* 142 */ NdrFcShort( 0x0 ), /* 0 */
/* 144 */ NdrFcShort( 0x8 ), /* 8 */
/* 146 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has return,
has ext, */
/* 148 */ 0xa, 0x3, /* 3 */
                                0x7, /* 10 */ /* Ext Flags: new corr desc, clt corr
check, srv corr check, */
/* 150 */ NdrFcShort( 0x20 ), /* 32 */
/* 152 */ NdrFcShort( 0x20 ), /* 32 */
/* 154 */ NdrFcShort( 0x0 ), /* 0 */
/* 156 */ NdrFcShort( 0x0 ), /* 0 */

/* Parameter txn_in */

/* 158 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifndef ALPHA
/* 160 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
                                NdrFcShort( 0x8 ), /* axp64 Stack size/offset = 8 */
#endif
/* 162 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */

/* Parameter txn_out */

/* 164 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=24 */
#ifndef ALPHA
/* 166 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
                                NdrFcShort( 0x20 ), /* axp64 Stack size/offset = 32 */
#endif
#endif
```

Appendix A - Application Source Code

```
/* 168 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Return value */
/* 170 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef ALPHA
/* 172 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
#else
NdrFcShort( 0x28 ), /* axp64 Stack size/offset = 40 */
#endif
/* 174 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure OrderStatus */
/* 176 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, Oi2 */
/* 178 */ NdrFcLong( 0x0 ), /* 0 */
/* 182 */ NdrFcShort( 0x7 ), /* 7 */
#ifdef ALPHA
/* 184 */ NdrFcShort( 0x38 ), /* ia64 Stack size/offset = 56 */
#else
NdrFcShort( 0x30 ), /* axp64 Stack size/offset = 48 */
#endif
/* 186 */ NdrFcShort( 0x0 ), /* 0 */
/* 188 */ NdrFcShort( 0x8 ), /* 8 */
/* 190 */ 0x47, /* Oi2 Flags: srv must size, clt must size, has return,
has ext, */
0x3, /* 3 */
/* 192 */ 0xa, /* 10 */
0x7, /* Ext Flags: new corr desc, clt corr
check, srv corr check, */
/* 194 */ NdrFcShort( 0x20 ), /* 32 */
/* 196 */ NdrFcShort( 0x20 ), /* 32 */
/* 198 */ NdrFcShort( 0x0 ), /* 0 */
/* 200 */ NdrFcShort( 0x0 ), /* 0 */
/* Parameter txn_in */
/* 202 */ NdrFcShort( 0x8b ), /* Flags: must size, must free, in, by val, */
#ifdef ALPHA
/* 204 */ NdrFcShort( 0x10 ), /* ia64 Stack size/offset = 16 */
#else
NdrFcShort( 0x8 ), /* axp64 Stack size/offset = 8 */
#endif
/* 206 */ NdrFcShort( 0x3b6 ), /* Type Offset=950 */
/* Parameter txn_out */
/* 208 */ NdrFcShort( 0x6113 ), /* Flags: must size, must free, out, simple ref,
srv alloc size=24 */
#ifdef ALPHA
/* 210 */ NdrFcShort( 0x28 ), /* ia64 Stack size/offset = 40 */
#else
NdrFcShort( 0x20 ), /* axp64 Stack size/offset = 32 */
#endif
/* 212 */ NdrFcShort( 0x3c8 ), /* Type Offset=968 */
/* Return value */
/* 214 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
#ifdef ALPHA
/* 216 */ NdrFcShort( 0x30 ), /* ia64 Stack size/offset = 48 */
```

```
#else
NdrFcShort( 0x28 ), /* axp64 Stack size/offset = 40 */
#endif
/* 218 */ 0x8, /* FC_LONG */
0x0, /* 0 */
/* Procedure CallSetComplete */
/* 220 */ 0x33, /* FC_AUTO_HANDLE */
0x6c, /* Old Flags: object, Oi2 */
/* 222 */ NdrFcLong( 0x0 ), /* 0 */
/* 226 */ NdrFcShort( 0x8 ), /* 8 */
/* 228 */ NdrFcShort( 0x10 ), /* ia64, axp64 Stack size/offset = 16 */
/* 230 */ NdrFcShort( 0x0 ), /* 0 */
/* 232 */ NdrFcShort( 0x8 ), /* 8 */
/* 234 */ 0x44, /* Oi2 Flags: has return, has ext, */
0x1, /* 1 */
/* 236 */ 0xa, /* 10 */
0x1, /* Ext Flags: new corr desc, */
/* 238 */ NdrFcShort( 0x0 ), /* 0 */
/* 240 */ NdrFcShort( 0x0 ), /* 0 */
/* 242 */ NdrFcShort( 0x0 ), /* 0 */
/* 244 */ NdrFcShort( 0x0 ), /* 0 */
/* Return value */
/* 246 */ NdrFcShort( 0x70 ), /* Flags: out, return, base type, */
/* 248 */ NdrFcShort( 0x8 ), /* ia64, axp64 Stack size/offset = 8 */
/* 250 */ 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( 0x39e ), /* Offset= 926 (930) */
/* 6 */
0x2b, /* FC_NON_ENCAPSULATED_UNION */
0x9, /* FC_ULONG */
/* 8 */ 0x7, /* Corr desc: FC_USHORT */
0x0, /* */
/* 10 */ NdrFcShort( 0xffff8 ), /* -8 */
/* 12 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 14 */ NdrFcShort( 0x2 ), /* Offset= 2 (16) */
/* 16 */ NdrFcShort( 0x10 ), /* 16 */
/* 18 */ NdrFcShort( 0x2b ), /* 43 */
/* 20 */ NdrFcLong( 0x3 ), /* 3 */
/* 24 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 26 */ NdrFcLong( 0x11 ), /* 17 */
/* 30 */ NdrFcShort( 0x8001 ), /* Simple arm type: FC_BYTE */
/* 32 */ NdrFcLong( 0x2 ), /* 2 */
/* 36 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 38 */ NdrFcLong( 0x4 ), /* 4 */
/* 42 */ NdrFcShort( 0x800a ), /* Simple arm type: FC_FLOAT */
/* 44 */ NdrFcLong( 0x5 ), /* 5 */
/* 48 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
```

Appendix A - Application Source Code

```
/* 50 */ NdrFcLong( 0xb ), /* 11 */
/* 54 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 56 */ NdrFcLong( 0xa ), /* 10 */
/* 60 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 62 */ NdrFcLong( 0x6 ), /* 6 */
/* 66 */ NdrFcShort( 0xd6 ), /* Offset= 214 (280) */
/* 68 */ NdrFcLong( 0x7 ), /* 7 */
/* 72 */ NdrFcShort( 0x800c ), /* Simple arm type: FC_DOUBLE */
/* 74 */ NdrFcLong( 0x8 ), /* 8 */
/* 78 */ NdrFcShort( 0xd0 ), /* Offset= 208 (286) */
/* 80 */ NdrFcLong( 0xd ), /* 13 */
/* 84 */ NdrFcShort( 0xe4 ), /* Offset= 228 (312) */
/* 86 */ NdrFcLong( 0x9 ), /* 9 */
/* 90 */ NdrFcShort( 0xf0 ), /* Offset= 240 (330) */
/* 92 */ NdrFcLong( 0x2000 ), /* 8192 */
/* 96 */ NdrFcShort( 0xfc ), /* Offset= 252 (348) */
/* 98 */ NdrFcLong( 0x24 ), /* 36 */
/* 102 */ NdrFcShort( 0x2f4 ), /* Offset= 756 (858) */
/* 104 */ NdrFcLong( 0x4024 ), /* 16420 */
/* 108 */ NdrFcShort( 0x2ee ), /* Offset= 750 (858) */
/* 110 */ NdrFcLong( 0x4011 ), /* 16401 */
/* 114 */ NdrFcShort( 0x2ec ), /* Offset= 748 (862) */
/* 116 */ NdrFcLong( 0x4002 ), /* 16386 */
/* 120 */ NdrFcShort( 0x2ea ), /* Offset= 746 (866) */
/* 122 */ NdrFcLong( 0x4003 ), /* 16387 */
/* 126 */ NdrFcShort( 0x2e8 ), /* Offset= 744 (870) */
/* 128 */ NdrFcLong( 0x4004 ), /* 16388 */
/* 132 */ NdrFcShort( 0x2e6 ), /* Offset= 742 (874) */
/* 134 */ NdrFcLong( 0x4005 ), /* 16389 */
/* 138 */ NdrFcShort( 0x2e4 ), /* Offset= 740 (878) */
/* 140 */ NdrFcLong( 0x400b ), /* 16395 */
/* 144 */ NdrFcShort( 0x2d2 ), /* Offset= 722 (866) */
/* 146 */ NdrFcLong( 0x400a ), /* 16394 */
/* 150 */ NdrFcShort( 0x2d0 ), /* Offset= 720 (870) */
/* 152 */ NdrFcLong( 0x4006 ), /* 16390 */
/* 156 */ NdrFcShort( 0x2d6 ), /* Offset= 726 (882) */
/* 158 */ NdrFcLong( 0x4007 ), /* 16391 */
/* 162 */ NdrFcShort( 0x2cc ), /* Offset= 716 (878) */
/* 164 */ NdrFcLong( 0x4008 ), /* 16392 */
/* 168 */ NdrFcShort( 0x2ce ), /* Offset= 718 (886) */
/* 170 */ NdrFcLong( 0x400d ), /* 16397 */
/* 174 */ NdrFcShort( 0x2cc ), /* Offset= 716 (890) */
/* 176 */ NdrFcLong( 0x4009 ), /* 16393 */
/* 180 */ NdrFcShort( 0x2ca ), /* Offset= 714 (894) */
/* 182 */ NdrFcLong( 0x6000 ), /* 24576 */
/* 186 */ NdrFcShort( 0x2c8 ), /* Offset= 712 (898) */
/* 188 */ NdrFcLong( 0x400c ), /* 16396 */
/* 192 */ NdrFcShort( 0x2c6 ), /* Offset= 710 (902) */
/* 194 */ NdrFcLong( 0x10 ), /* 16 */
/* 198 */ NdrFcShort( 0x8002 ), /* Simple arm type: FC_CHAR */
/* 200 */ NdrFcLong( 0x12 ), /* 18 */
/* 204 */ NdrFcShort( 0x8006 ), /* Simple arm type: FC_SHORT */
/* 206 */ NdrFcLong( 0x13 ), /* 19 */
/* 210 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 212 */ NdrFcLong( 0x16 ), /* 22 */
/* 216 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 218 */ NdrFcLong( 0x17 ), /* 23 */
/* 222 */ NdrFcShort( 0x8008 ), /* Simple arm type: FC_LONG */
/* 224 */ NdrFcLong( 0xe ), /* 14 */
/* 228 */ NdrFcShort( 0x2aa ), /* Offset= 682 (910) */
/* 230 */ NdrFcLong( 0x400e ), /* 16398 */
/* 234 */ NdrFcShort( 0x2b0 ), /* Offset= 688 (922) */
/* 236 */ NdrFcLong( 0x4010 ), /* 16400 */

/* 240 */ NdrFcShort( 0x2ae ), /* Offset= 686 (926) */
/* 242 */ NdrFcLong( 0x4012 ), /* 16402 */
/* 246 */ NdrFcShort( 0x26c ), /* Offset= 620 (866) */
/* 248 */ NdrFcLong( 0x4013 ), /* 16403 */
/* 252 */ NdrFcShort( 0x26a ), /* Offset= 618 (870) */
/* 254 */ NdrFcLong( 0x4016 ), /* 16406 */
/* 258 */ NdrFcShort( 0x264 ), /* Offset= 612 (870) */
/* 260 */ NdrFcLong( 0x4017 ), /* 16407 */
/* 264 */ NdrFcShort( 0x25e ), /* Offset= 606 (870) */
/* 266 */ NdrFcLong( 0x0 ), /* 0 */
/* 270 */ NdrFcShort( 0x0 ), /* Offset= 0 (270) */
/* 272 */ NdrFcLong( 0x1 ), /* 1 */
/* 276 */ NdrFcShort( 0x0 ), /* Offset= 0 (276) */
/* 278 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (277) */
/* 280 */

/* 282 */ NdrFcShort( 0x8 ), /* 8 */
/* 284 */ 0xb, /* FC_HYPER */
/* 286 */ 0x5b, /* FC_END */

/* 288 */ NdrFcShort( 0xe ), /* FC_UP */
/* 290 */ 0x12, 0x0, /* Offset= 14 (302) */

/* 292 */ NdrFcShort( 0x2 ), /* 2 */
/* 294 */ 0x9, /* Corr desc: FC_ULONG */
/* 296 */ NdrFcShort( 0xfffc ), /* -4 */
/* 298 */ NdrFcShort( 0x1 ), /* Corr flags: early */
/* 300 */ 0x6, /* FC_SHORT */
/* 302 */ 0x5b, /* FC_END */

/* 304 */ NdrFcShort( 0x8 ), /* 8 */
/* 306 */ NdrFcShort( 0xfffffff0 ), /* Offset= -16 (290) */
/* 308 */ 0x8, /* FC_LONG */
/* 310 */ 0x5c, /* FC_PAD */
/* 312 */ 0x5b, /* FC_END */

/* 314 */ NdrFcLong( 0x0 ), /* 0 */
/* 318 */ NdrFcShort( 0x0 ), /* 0 */
/* 320 */ NdrFcShort( 0x0 ), /* 0 */
/* 322 */ 0xc0, /* 192 */
/* 324 */ 0x0, /* 0 */
/* 326 */ 0x0, /* 0 */
/* 328 */ 0x0, /* 0 */
/* 330 */ 0x46, /* 70 */

/* 332 */ NdrFcLong( 0x20400 ), /* 132096 */
/* 336 */ NdrFcShort( 0x0 ), /* 0 */
/* 338 */ NdrFcShort( 0x0 ), /* 0 */
/* 340 */ 0xc0, /* 192 */

0x15, /* FC_STRUCT */
0x7, /* 7 */
0x5b, /* FC_END */
0x12, 0x0, /* FC_UP */
0x1b, /* FC_CARRAY */
0x1, /* 1 */
0x2, /* 2 */
0x9, /* Corr desc: FC_ULONG */
0x0, /* 0 */
0xfffc, /* -4 */
0x1, /* Corr flags: early */
0x6, /* FC_SHORT */
0x5b, /* FC_END */
0x17, /* FC_CSTRUCT */
0x3, /* 3 */
0x8, /* 8 */
0xfffffff0, /* Offset= -16 (290) */
0x8, /* FC_LONG */
0x5c, /* FC_PAD */
0x5b, /* FC_END */
0x2f, /* FC_IP */
0x5a, /* FC_CONSTANT_IID */
0x0, /* 0 */
0x0, /* 0 */
0x0, /* 0 */
0x0, /* 0 */
0x0, /* 0 */
0x46, /* 70 */
0x2f, /* FC_IP */
0x5a, /* FC_CONSTANT_IID */
0x20400, /* 132096 */
0x0, /* 0 */
0x0, /* 0 */
0xc0, /* 192 */
```

Appendix A - Application Source Code

```

0x0, /* 0 */
/* 342 */ 0x0, /* 0 */
0x0, /* 0 */
/* 344 */ 0x0, /* 0 */
/* 346 */ 0x0, /* 0 */
0x46, /* 70 */
/* 348 */
0x12, 0x10, /* FC_UP [pointer_deref] */
/* 350 */ NdrFcShort( 0x2 ), /* Offset= 2 (352) */
/* 352 */
0x12, 0x0, /* FC_UP */
/* 354 */ NdrFcShort( 0x1e6 ), /* Offset= 486 (840) */
/* 356 */
0x2a, /* FC_ENCAPSULATED_UNION */
0x89, /* 137 */
/* 358 */ NdrFcShort( 0x20 ), /* 32 */
/* 360 */ NdrFcShort( 0xa ), /* 10 */
/* 362 */ NdrFcLong( 0x8 ), /* 8 */
/* 366 */ NdrFcShort( 0x50 ), /* Offset= 80 (446) */
/* 368 */ NdrFcLong( 0xd ), /* 13 */
/* 372 */ NdrFcShort( 0x70 ), /* Offset= 112 (484) */
/* 374 */ NdrFcLong( 0x9 ), /* 9 */
/* 378 */ NdrFcShort( 0x90 ), /* Offset= 144 (522) */
/* 380 */ NdrFcLong( 0xc ), /* 12 */
/* 384 */ NdrFcShort( 0xb0 ), /* Offset= 176 (560) */
/* 386 */ NdrFcLong( 0x24 ), /* 36 */
/* 390 */ NdrFcShort( 0x104 ), /* Offset= 260 (650) */
/* 392 */ NdrFcLong( 0x800d ), /* 32781 */
/* 396 */ NdrFcShort( 0x120 ), /* Offset= 288 (684) */
/* 398 */ NdrFcLong( 0x10 ), /* 16 */
/* 402 */ NdrFcShort( 0x13a ), /* Offset= 314 (716) */
/* 404 */ NdrFcLong( 0x2 ), /* 2 */
/* 408 */ NdrFcShort( 0x150 ), /* Offset= 336 (744) */
/* 410 */ NdrFcLong( 0x3 ), /* 3 */
/* 414 */ NdrFcShort( 0x166 ), /* Offset= 358 (772) */
/* 416 */ NdrFcLong( 0x14 ), /* 20 */
/* 420 */ NdrFcShort( 0x17c ), /* Offset= 380 (800) */
/* 422 */ NdrFcShort( 0xffffffff ), /* Offset= -1 (421) */
/* 424 */
0x21, /* FC_BOGUS_ARRAY */
0x3, /* 3 */
/* 426 */ NdrFcShort( 0x0 ), /* 0 */
/* 428 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 430 */ NdrFcShort( 0x0 ), /* 0 */
/* 432 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 434 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 438 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 440 */
0x12, 0x0, /* FC_UP */
/* 442 */ NdrFcShort( 0xffffffff74 ), /* Offset= -140 (302) */
/* 444 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 446 */
0x1a, /* FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 448 */ NdrFcShort( 0x10 ), /* 16 */
/* 450 */ NdrFcShort( 0x0 ), /* 0 */
/* 452 */ NdrFcShort( 0x6 ), /* Offset= 6 (458) */
/* 454 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 456 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
0x5b, /* FC_END */
0x11, 0x0, /* FC_RP */
/* 460 */ NdrFcShort( 0xffffffffdc ), /* Offset= -36 (424) */
/* 462 */
0x21, /* FC_BOGUS_ARRAY */
0x3, /* 3 */
/* 464 */ NdrFcShort( 0x0 ), /* 0 */
/* 466 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 468 */ NdrFcShort( 0x0 ), /* 0 */
/* 470 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 472 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 476 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 478 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 480 */ NdrFcShort( 0xffffffff58 ), /* Offset= -168 (312) */
/* 482 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 484 */
0x1a, /* FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 486 */ NdrFcShort( 0x10 ), /* 16 */
/* 488 */ NdrFcShort( 0x0 ), /* 0 */
/* 490 */ NdrFcShort( 0x6 ), /* Offset= 6 (496) */
/* 492 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 494 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 496 */
0x11, 0x0, /* FC_RP */
/* 498 */ NdrFcShort( 0xffffffffdc ), /* Offset= -36 (462) */
/* 500 */
0x21, /* FC_BOGUS_ARRAY */
0x3, /* 3 */
/* 502 */ NdrFcShort( 0x0 ), /* 0 */
/* 504 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
0x0, /* */
/* 506 */ NdrFcShort( 0x0 ), /* 0 */
/* 508 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 510 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 514 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 516 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
0x0, /* 0 */
/* 518 */ NdrFcShort( 0xffffffff44 ), /* Offset= -188 (330) */
/* 520 */ 0x5c, /* FC_PAD */
0x5b, /* FC_END */
/* 522 */
0x1a, /* FC_BOGUS_STRUCT */
0x3, /* 3 */
/* 524 */ NdrFcShort( 0x10 ), /* 16 */
/* 526 */ NdrFcShort( 0x0 ), /* 0 */
/* 528 */ NdrFcShort( 0x6 ), /* Offset= 6 (534) */
/* 530 */ 0x8, /* FC_LONG */
0x39, /* FC_ALIGNM8 */
/* 532 */ 0x36, /* FC_POINTER */
0x5b, /* FC_END */
/* 534 */
0x11, 0x0, /* FC_RP */
/* 536 */ NdrFcShort( 0xffffffffdc ), /* Offset= -36 (500) */
/* 538 */
0x21, /* FC_BOGUS_ARRAY */
0x3, /* 3 */
```

Appendix A - Application Source Code

```
/* 540 */ NdrFcShort( 0x0 ), /* 0 */
/* 542 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
/* 544 */ NdrFcShort( 0x0 ), /* 0 */
/* 546 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 548 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 552 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 554 */
/* 556 */ NdrFcShort( 0x176 ), /* FC_UP */
/* 558 */ 0x5c, /* FC_PAD */
/* 560 */
/* 562 */ NdrFcShort( 0x10 ), /* 16 */
/* 564 */ NdrFcShort( 0x0 ), /* 0 */
/* 566 */ NdrFcShort( 0x6 ), /* Offset= 6 (572) */
/* 568 */ 0x8, /* FC_LONG */
/* 570 */ 0x36, /* FC_POINTER */
/* 572 */
/* 574 */ NdrFcShort( 0xffffffffc ), /* Offset= -36 (538) */
/* 576 */
/* 578 */ NdrFcLong( 0x2f ), /* FC_IP */
/* 582 */ NdrFcShort( 0x0 ), /* FC_CONSTANT_IID */
/* 584 */ NdrFcShort( 0x0 ), /* 47 */
/* 586 */ 0xc0, /* 0 */
/* 588 */ 0x0, /* 0 */
/* 590 */ 0x0, /* 0 */
/* 592 */ 0x0, /* 0 */
/* 594 */
/* 596 */ NdrFcShort( 0x1 ), /* FC_CARRAY */
/* 598 */ 0x19, /* 1 */
/* 600 */ NdrFcShort( 0x4 ), /* Corr desc: field pointer, FC_ULONG */
/* 602 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 604 */ 0x1, /* FC_BYTE */
/* 606 */
/* 608 */ NdrFcShort( 0x18 ), /* FC_BOGUS_STRUCT */
/* 610 */ NdrFcShort( 0x0 ), /* 3 */
/* 612 */ NdrFcShort( 0xc ), /* Offset= 12 (624) */
/* 614 */ 0x8, /* FC_LONG */
/* 616 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 618 */ NdrFcShort( 0xffffffffd6 ), /* Offset= -42 (576) */
/* 620 */ 0x39, /* FC_ALIGNM8 */
/* 622 */ 0x5c, /* FC_POINTER */
/* FC_PAD */
/* 624 */
/* 626 */ NdrFcShort( 0xffffffe0 ), /* FC_UP */
/* 628 */
/* 630 */ NdrFcShort( 0x0 ), /* FC_BOGUS_ARRAY */
/* 632 */ 0x19, /* 3 */
/* 634 */ NdrFcShort( 0x0 ), /* 0 */
/* 636 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 638 */ NdrFcLong( 0xffffffff ), /* -1 */
/* 642 */ NdrFcShort( 0x0 ), /* Corr flags: */
/* 644 */
/* 646 */ NdrFcShort( 0xffffffd8 ), /* FC_UP */
/* 648 */ 0x5c, /* Offset= -40 (606) */
/* FC_PAD */
/* 650 */
/* 652 */ NdrFcShort( 0x10 ), /* FC_BOGUS_STRUCT */
/* 654 */ NdrFcShort( 0x0 ), /* 3 */
/* 656 */ NdrFcShort( 0x6 ), /* Offset= 6 (662) */
/* 658 */ 0x8, /* FC_LONG */
/* 660 */ 0x36, /* FC_ALIGNM8 */
/* FC_POINTER */
/* 662 */
/* 664 */ NdrFcShort( 0xffffffffc ), /* FC_UP */
/* 666 */
/* 668 */ NdrFcShort( 0x8 ), /* FC_SMFARRAY */
/* 670 */ 0x2, /* 8 */
/* FC_CHAR */
/* 672 */
/* 674 */ NdrFcShort( 0x10 ), /* FC_STRUCT */
/* 676 */ 0x8, /* 3 */
/* FC_LONG */
/* 678 */ 0x6, /* FC_SHORT */
/* 680 */ 0x0, /* FC_EMBEDDED_COMPLEX */
/* 684 */
/* 686 */ NdrFcShort( 0xfffffff1 ), /* Offset= -15 (666) */
/* 688 */ 0x5b, /* FC_END */
/* 690 */
/* 692 */ NdrFcShort( 0x10 ), /* FC_BOGUS_STRUCT */
/* 694 */ 0x36, /* 3 */
/* 696 */ 0x0, /* FC_LONG */
/* 698 */ 0x8, /* Offset= 10 (700) */
/* 700 */ 0x39, /* FC_ALIGNM8 */
/* FC_POINTER */
/* 702 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
/* 704 */ 0x0, /* FC_POINTER */
/* 706 */ NdrFcShort( 0xffffffe7 ), /* Offset= -25 (672) */
/* 708 */ 0x5b, /* FC_END */
/* 710 */
```

Appendix A - Application Source Code

```
/* 702 */ NdrFcShort( 0xffffffff10 ), /* Offset= -240 (462) */
/* 704 */
    0x1b, /* FC_CARRAY */
    0x0, /* 0 */
/* 706 */ NdrFcShort( 0x1 ), /* 1 */
/* 708 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
    0x0, /* */
/* 710 */ NdrFcShort( 0x0 ), /* 0 */
/* 712 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 714 */ 0x1, /* FC_BYTE */
    0x5b, /* FC_END */
/* 716 */
    0x1a, /* FC_BOGUS_STRUCT */
    0x3, /* 3 */
/* 718 */ NdrFcShort( 0x10 ), /* 16 */
/* 720 */ NdrFcShort( 0x0 ), /* 0 */
/* 722 */ NdrFcShort( 0x6 ), /* Offset= 6 (728) */
/* 724 */ 0x8, /* FC_LONG */
    0x39, /* FC_ALIGNM8 */
/* 726 */ 0x36, /* FC_POINTER */
    0x5b, /* FC_END */
/* 728 */
    0x12, 0x0, /* FC_UP */
/* 730 */ NdrFcShort( 0xffffffffe6 ), /* Offset= -26 (704) */
/* 732 */
    0x1b, /* FC_CARRAY */
    0x1, /* 1 */
/* 734 */ NdrFcShort( 0x2 ), /* 2 */
/* 736 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
    0x0, /* */
/* 738 */ NdrFcShort( 0x0 ), /* 0 */
/* 740 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 742 */ 0x6, /* FC_SHORT */
    0x5b, /* FC_END */
/* 744 */
    0x1a, /* FC_BOGUS_STRUCT */
    0x3, /* 3 */
/* 746 */ NdrFcShort( 0x10 ), /* 16 */
/* 748 */ NdrFcShort( 0x0 ), /* 0 */
/* 750 */ NdrFcShort( 0x6 ), /* Offset= 6 (756) */
/* 752 */ 0x8, /* FC_LONG */
    0x39, /* FC_ALIGNM8 */
/* 754 */ 0x36, /* FC_POINTER */
    0x5b, /* FC_END */
/* 756 */
    0x12, 0x0, /* FC_UP */
/* 758 */ NdrFcShort( 0xffffffffe6 ), /* Offset= -26 (732) */
/* 760 */
    0x1b, /* FC_CARRAY */
    0x3, /* 3 */
/* 762 */ NdrFcShort( 0x4 ), /* 4 */
/* 764 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
    0x0, /* */
/* 766 */ NdrFcShort( 0x0 ), /* 0 */
/* 768 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 770 */ 0x8, /* FC_LONG */
    0x5b, /* FC_END */
/* 772 */
    0x1a, /* FC_BOGUS_STRUCT */
    0x3, /* 3 */
/* 774 */ NdrFcShort( 0x10 ), /* 16 */
/* 776 */ NdrFcShort( 0x0 ), /* 0 */
    0x6, /* Offset= 6 (784) */
/* 778 */ NdrFcShort( 0x6 ), /* FC_LONG */
    0x39, /* FC_ALIGNM8 */
/* 782 */ 0x36, /* FC_POINTER */
    0x5b, /* FC_END */
/* 784 */
    0x12, 0x0, /* FC_UP */
/* 786 */ NdrFcShort( 0xffffffffe6 ), /* Offset= -26 (760) */
/* 788 */
    0x1b, /* FC_CARRAY */
    0x7, /* 7 */
/* 790 */ NdrFcShort( 0x8 ), /* 8 */
/* 792 */ 0x19, /* Corr desc: field pointer, FC_ULONG */
    0x0, /* */
/* 794 */ NdrFcShort( 0x0 ), /* 0 */
/* 796 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 798 */ 0xb, /* FC_HYPER */
    0x5b, /* FC_END */
/* 800 */
    0x1a, /* FC_BOGUS_STRUCT */
    0x3, /* 3 */
/* 802 */ NdrFcShort( 0x10 ), /* 16 */
/* 804 */ NdrFcShort( 0x0 ), /* 0 */
/* 806 */ NdrFcShort( 0x6 ), /* Offset= 6 (812) */
/* 808 */ 0x8, /* FC_LONG */
    0x39, /* FC_ALIGNM8 */
/* 810 */ 0x36, /* FC_POINTER */
    0x5b, /* FC_END */
/* 812 */
    0x12, 0x0, /* FC_UP */
/* 814 */ NdrFcShort( 0xffffffffe6 ), /* Offset= -26 (788) */
/* 816 */
    0x15, /* FC_STRUCT */
    0x3, /* 3 */
/* 818 */ NdrFcShort( 0x8 ), /* 8 */
/* 820 */ 0x8, /* FC_LONG */
    0x8, /* FC_LONG */
    0x5b, /* FC_PAD */
    0x5b, /* FC_END */
/* 824 */
    0x1b, /* FC_CARRAY */
    0x3, /* 3 */
/* 826 */ NdrFcShort( 0x8 ), /* 8 */
/* 828 */ 0x7, /* Corr desc: FC_USHORT */
    0x0, /* */
/* 830 */ NdrFcShort( 0xffc8 ), /* -56 */
/* 832 */ NdrFcShort( 0x1 ), /* Corr flags: early, */
/* 834 */ 0x4c, /* FC_EMBEDDED_COMPLEX */
    0x0, /* 0 */
/* 836 */ NdrFcShort( 0xffffffffec ), /* Offset= -20 (816) */
/* 838 */ 0x5c, /* FC_PAD */
    0x5b, /* FC_END */
/* 840 */
    0x1a, /* FC_BOGUS_STRUCT */
    0x3, /* 3 */
/* 842 */ NdrFcShort( 0x38 ), /* 56 */
/* 844 */ NdrFcShort( 0xffffffffec ), /* Offset= -20 (824) */
/* 846 */ NdrFcShort( 0x0 ), /* Offset= 0 (846) */
/* 848 */ 0x6, /* FC_SHORT */
    0x6, /* FC_SHORT */
/* 850 */ 0x38, /* FC_ALIGNM4 */
    0x8, /* FC_LONG */
/* 852 */ 0x8, /* FC_LONG */
```

Appendix A - Application Source Code

```
/* 854 */ Oxd4, /* FC_EMBEDDED_COMPLEX */
/* 4 */
NdrFcShort( 0xfffffe0d ), /* Offset= -499 (356) */
/* 858 */ /* FC_END */

/* 860 */ 0x12, 0x0, /* FC_UP */
NdrFcShort( 0xfffff02 ), /* Offset= -254 (606) */
/* 862 */

/* 864 */ 0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_BYTE */
/* 866 */ /* FC_PAD */
0x5c,

/* 868 */ 0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_SHORT */
/* 870 */ /* FC_PAD */
0x5c,

/* 872 */ 0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_LONG */
/* 874 */ /* FC_PAD */
0x5c,

/* 876 */ 0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_FLOAT */
/* 878 */ /* FC_PAD */
0x5c,

/* 880 */ 0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_DOUBLE */
/* 882 */ /* FC_PAD */
0x5c,

/* 884 */ 0x12, 0x0, /* FC_UP */
NdrFcShort( 0xfffffd4 ), /* Offset= -604 (280) */
/* 886 */

/* 888 */ 0x12, 0x10, /* FC_UP [pointer_deref] */
NdrFcShort( 0xfffffa6 ), /* Offset= -602 (286) */
/* 890 */

/* 892 */ 0x12, 0x10, /* FC_UP [pointer_deref] */
NdrFcShort( 0xfffffdbc ), /* Offset= -580 (312) */
/* 894 */

/* 896 */ 0x12, 0x10, /* FC_UP [pointer_deref] */
NdrFcShort( 0xfffffdca ), /* Offset= -566 (330) */
/* 898 */

/* 900 */ 0x12, 0x10, /* FC_UP [pointer_deref] */
NdrFcShort( 0xffffdd8 ), /* Offset= -552 (348) */
/* 902 */

/* 904 */ 0x12, 0x10, /* FC_UP [pointer_deref] */
NdrFcShort( 0x2 ), /* Offset= 2 (906) */
/* 906 */

/* 908 */ 0x12, 0x0, /* FC_UP */
NdrFcShort( 0x16 ), /* Offset= 22 (930) */
/* 910 */

/* 912 */ 0x15, /* FC_STRUCT */
/* 7 */
/* 914 */ 0x7, /* FC_UP */
/* 16 */
/* 916 */ 0x6, /* FC_SHORT */
/* FC_BYTE */
/* 918 */ 0x1, /* FC_BYTE */
/* FC_ALIGNM4 */
/* 920 */ 0x8, /* FC_LONG */
/* FC_ALIGNM8 */
/* 922 */ 0x39, /* FC_HYPER */
/* FC_END */
0x5b,

/* 924 */ 0x12, 0x0, /* FC_UP */
NdrFcShort( 0xffffff2 ), /* Offset= -14 (910) */
```

```
/* 926 */
/* 928 */ 0x2, 0x12, 0x8, /* FC_UP [simple_pointer] */
/* FC_CHAR */
/* 930 */ 0x5c, /* FC_PAD */

/* 932 */ 0x1a, /* FC_BOGUS_STRUCT */
/* 7 */
/* 934 */ 0x7, /* FC_UP */
/* 32 */
/* 936 */ 0x0, /* FC_UP */
/* 0 */
/* 938 */ 0x8, /* FC_UP [simple_pointer] */
/* Offset= 0 (936) */
/* FC_LONG */
/* 940 */ 0x8, /* FC_LONG */
/* FC_SHORT */
/* 942 */ 0x6, /* FC_SHORT */
/* FC_SHORT */
/* 944 */ 0x6, /* FC_SHORT */
/* FC_SHORT */
/* FC_EMBEDDED_COMPLEX */
/* 0 */
/* 946 */ 0x0, /* FC_UP */
/* Offset= -940 (6) */
/* 948 */ 0x5c, /* FC_PAD */
/* FC_END */
/* 950 */ 0xb4, /* FC_USER_MARSHAL */
/* FC_USER_MARSHAL */
/* 131 */
/* 952 */ 0x83, /* FC_UP */
/* 0 */
/* 954 */ 0x0, /* FC_UP */
/* 24 */
/* 956 */ 0x18, /* FC_UP */
/* 0 */
/* 958 */ 0x0, /* FC_UP */
/* Offset= -956 (2) */
/* 960 */

/* 962 */ 0x11, 0x4, /* FC_UP [allocated_on_stack] */
NdrFcShort( 0x6 ), /* Offset= 6 (968) */
/* 964 */

/* 966 */ 0x13, 0x0, /* FC_UP [allocated_on_stack] */
NdrFcShort( 0xfffffddc ), /* Offset= -36 (930) */
/* 968 */ 0xb4, /* FC_USER_MARSHAL */
/* FC_USER_MARSHAL */
/* 131 */
/* 970 */ 0x83, /* FC_UP */
/* 0 */
/* 972 */ 0x0, /* FC_UP */
/* 24 */
/* 974 */ 0x18, /* FC_UP */
/* 0 */
/* 976 */ 0x0, /* FC_UP */
/* Offset= -12 (964) */

}

};

const CInterfaceProxyVtbl * _tpcc_com_ps_ProxyVtblList[] =
{
    ( CInterfaceProxyVtbl *) &ITPCCProxyVtbl,
    0
};

const CInterfaceStubVtbl * _tpcc_com_ps_StubVtblList[] =
{
    ( CInterfaceStubVtbl *) &ITPCCStubVtbl,
    0
};

PCInterfaceName const _tpcc_com_ps_InterfaceNamesList[] =
{
    "ITPCC",
    0
};
```

Appendix A - Application Source Code

```
#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_uid] interfaces */
    0, /* Filler1 */
    0, /* Filler2 */
    0 /* Filler3 */
};

#endif /* defined(_M_IA64) || defined(_M_AXP64)*/
```

common/txnlog/include/rvertime.h

```
/* FILE: rvertime.h : header file
 * Copyright 1997 Microsoft Corp., All rights reserved.
 *
 * Authors: Charles Levine, Philip Durr
 *          Microsoft Corp.
 */

#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.
 *
 * Authors: Mike Parkes, Charles Levine, Philip Durr
 *          Microsoft Corp.
 */

#ifndef _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 builds.
    volatile LONG TotalLocks;
    volatile LONG TotalSleeps;
    volatile LONG TotalSpins;
    volatile LONG TotalWaits;
#endif

public:
    // Public functions.
    Spinlock( void );
};
```


Appendix A - Application Source Code

```
        inline BOOL ClaimLock( BOOL Wait = TRUE );
        inline void ReleaseLock( void );
        ~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
 *
 * NOTE: this file is RTE specific and should not be
 * included in Full Disclosure Reports.
 *
 * Copyright Microsoft, 1999
 *
 * PURPOSE: Structure definitions for logging delivery txn completion stats.
 * Contact: Charles Levine (clevine@microsoft.com)
 */

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 //
```

Appendix A - Application Source Code

```

#define TXN_REC_TYPE_TPCC                2        // replaces
TRANSACTION_TYPE_TPCC
#define TXN_REC_TYPE_TPCC_DELIV_DEF     3

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;               // menu time (ms)

```

```

    int          DeltaT2;               // keying time (ms)
    int          DeltaT3;               // think time (ms)
    int          DeltaT4;               // response time (ms)
    int          RTDelay;               // response time delay (ms)
    int          TxnError;              // error code providing more
detail for TxnStatus
    WORD         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;             // completion status for txn
to indicate errors
    BYTE         reserved;              // for word alignment
    TXN_DETAILS   TxnDetails;           //
} 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)
    WORD         w_id;                  // warehouse ID
    BYTE         TxnStatus;             // completion status for txn
to indicate errors
    BYTE         reserved;              // for word alignment
    short        o_carrier_id;          // carrier id
    long         o_id[10];              // returned delivery transaction ids
} TXN_RECORD_TPCC_DELIV_DEF, *PTXN_RECORD_TPCC_DELIV_DEF;

#define TXN_LOG_VERSION                  1
#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

```

Appendix A - Application Source Code

```

        JULIAN_TIME          EndTxnTS;          // timestamp of
last (highest) txn completion time
        int
        // number of records in log file
        BOOL                iRecCount;
        // file size in bytes
        int                bLogSorted;
        // the record map provides a fast way to get close to a particular
        // timestamp in a sorted log file.
        struct
        {
            JULIAN_TIME      TS;
            // timestamp of record
            int              iPos;
            // byte position in file
        }
        RecMap[RecMapSize];
#define RecMapSize 200
    } TXN_LOG_HEADER, *PTXN_LOG_HEADER;

#define READ_BUFFER_SIZE 64*1024
#define WRITE_BUFFER_SIZE 8*1024

#define NUM_READ_BUFFERS 1
#define NUM_WRITE_BUFFERS 2
#define MAX_NUM_BUFFERS 2

// flags passed in to the constructor
#define TXN_LOG_WRITE 0x01
#define TXN_LOG_READ 0x02
#define TXN_LOG_SORTED 0x04

#define TXN_LOG_OS_ERROR 1
#define TXN_LOG_NOT_SORTED 2

#define SKIP_CTRL_RECS 1

class CTxnLog
{
private:
    DWORD iBufferSize; //buffer
    allocated size
    DWORD iBytesFreeInBuffer; //total bytes
    available for use in buffer
    int iNumBuffers;
    //buffers in use
    int iActiveBuffer;
    //indicates which buffer is active: 0 or 1
    int iIoBuffer;
    //buffer for any pending IO operation
    int iFilePointer;
    //position in file.
    int iNextRec;
    //when reading, ordinal value of next record

    // A "save point" is remembered each time GetNextRecord is called
    // with a start time specified.
    // The next time it is called, if start time is after the save point,
    // we start scanning from the

```

```

        // save point. This is particularly useful in FindBestInterval,
        // where the log is scanned repeatedly.
        JULIAN_TIME SavePtTime;
        int iSavePtFilePointer;
        int iSavePtNextRec;

        JULIAN_TIME lastTS;
        //when writing sorted output, used to verify records are sorted
        BOOL bWrite;
        //writing log file

        BOOL bLogSorted;
        // is log file sorted? applies to both input and output
        JULIAN_TIME BeginTxnTS;
        // timestamp of first (lowest) txn start
        JULIAN_TIME EndTxnTS; //
timestamp of last (highest) txn completion time
        int iRecCount;
        // number of records in log file

        BYTE *pCurrent;
        //ptr to current buffer
        BYTE *pBuffer[MAX_NUM_BUFFERS];

        PTXN_RECORD_HEADER *TxnArray; //transaction
        record pointer array for sort

        DWORD dwError;
        HANDLE hTxnFile; //handle
        to log file
        HANDLE hMapFile; //map
        file used when sorting the log
        HANDLE hIoComplete; //event
        to signify that there are no pending IOs
        HANDLE hLogFileIo;
        //event to signal the IO thread to write the inactive buffer

        Spinlock Spin; //spin
        lock to protect the txn log file buffers

        int Write(BYTE *ptr, DWORD Size);
        static void LogFileIO(CTxnLog *);

public:
    CTxnLog::CTxnLog(LPCTSTR szFileName, DWORD dwOpts);
    ~CTxnLog(void);

    int WriteToLog(PTXN_RECORD_TPCC pTxnRcld);
    int WriteToLog(PTXN_RECORD_TPCC_DELIV_DEF pTxnRcld);
    int WriteToLog(PTXN_RECORD_CONTROL pCtrlRec);
    int WriteToLog(PTXN_RECORD_HEADER pCtrlRec);

    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(int index);

```

Appendix A - Application Source Code

```
inline BOOL IsSorted(void) { return bLogSorted; };
inline JULIAN_TIME BeginTS(void) { return BeginTxnTS; };
inline JULIAN_TIME EndTS(void) { return EndTxnTS; };
inline int ReCordCount(void) { return iRecCount; };
};

class CTXNLOG_ERR : public CBaseErr
{
public:
    enum CTPCC_DBLIB_ERRS
    {
        ERR_BAD_FILE_FORMAT = 1,      // "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 ) { m_errno = iErr; };

    int          m_errno;

    int ErrorType() {return ERR_TYPE_TXNLOG;};
    int ErrorNum() {return m_errno;};

    // TODO: need to complete...
    char *ErrorText() {return "";};
};
```

Appendix B - Database Design

Appendix B - Database Design

Build Scripts

setup.cmd

```
ECHO OFF

@ECHO *****
@ECHO *
@ECHO * Microsoft TPC-C Benchmark Kit Ver. 4.01 *
@ECHO *
@ECHO *****

if '%1'==' ' goto usage
if '%2'==' ' goto usage
if '%3'==' ' goto usage
if '%4'==' ' goto usage
if not '%5'==' ' if not '%5' == 'scaled' goto usage

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

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

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

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

:bulddb
@if exist logs\db.log del logs\db.log >nul
@ECHO Building database files and database...
isql -Usa -P -S%1 -e < scripts\%2.war\%4\createdb.sql >
logs\db.log
@ECHO Database build complete.
if '%3'=='full' goto objects
```

```
goto end

:objects
@if exist logs\objects.log del logs\objects.log >nul
@ECHO Creating database objects...
isql -Usa -P -S%1 -e < scripts\ddl\%4\tables.sql > logs\objects.log
isql -Usa -P -S%1 -e < scripts\dml\%4\neword.sql >> logs\objects.log
isql -Usa -P -S%1 -e < scripts\dml\%4\payment.sql >> logs\objects.log
isql -Usa -P -S%1 -e < scripts\dml\%4\ordstat.sql >> logs\objects.log
isql -Usa -P -S%1 -e < scripts\dml\%4\delivery.sql >>
logs\objects.log
isql -Usa -P -S%1 -e < scripts\dml\%4\stocklev.sql >>
logs\objects.log
@ECHO Database object creation complete.
if '%3'=='full' goto bulkload
if '%3'=='objectsfull' goto bulkload
goto end

:bulkload
@if exist logs\bulkload.log del logs\bulkload.log >nul
@ECHO Beginning data load and index creation...
isql -Usa -P -S%1 -e < scripts\utility\%4\dbopt1.sql >>
logs\objects.log
if '%4'=='mssql70' goto odbc
if '%4'=='mssql65' goto dlib
goto usage
:dlib
if '%5'==' ' loader\%4\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -dscrip\ddl\%4 -c0
if '%5'=='normal' loader\%4\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -dscrip\ddl\%4 -c0
if '%5'=='scaled' loader\%4\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -dscrip\ddl\%4 -c1
goto bulkloaddone
:odbc
if '%5'==' ' loader\%4\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -dscrip\ddl\%4 -c0
if '%5'=='normal' loader\%4\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -dscrip\ddl\%4 -c0
if '%5'=='scaled' loader\%4\bin\tpccldr -S%1 -W%2 -flogs\bulkload.log -dscrip\ddl\%4 -c1
goto bulkloaddone
:bulkloaddone
isql -Usa -P -S%1 -e < scripts\utility\%4\dbopt2.sql >>
logs\bulkload.log
@ECHO Data load and index creation complete.
if '%3'=='full' goto backup
if '%3'=='objectsfull' goto backup
if '%3'=='bulkloadfull' goto backup
goto end

:backup
@if exist logs\backup.log del logs\backup.log >nul
@ECHO Backing up database...
isql -Usa -P -S%1 -e < scripts\%2.war\%4\backup.sql >
logs\backup.log
@ECHO Database backup complete.
if '%3'=='full' goto verifyload
if '%3'=='objectsfull' goto verifyload
if '%3'=='bulkloadfull' goto verifyload
goto complete

:verifyload
@if exist logs\verifyload.log del logs\verifyload.log >nul
@Echo Verifying TPC-C database load...
```

Appendix B - Database Design

```
isql -Usa -P -S%1 < scripts\utility\%4\verifytpccload.sql >
logs\verifyload.log
@ECHO Check logs\verifyload.log to verify database load.

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

goto end

:usage
@ECHO *****
@ECHO *
@ECHO * The TPC-C setup command file requires the following parameters: *
@ECHO *
@ECHO * setup SERVER NUMWAR BLDOPT VERSION DBTYPE *
@ECHO *
@ECHO * SERVER = machine name of server (use "" for local server) *
@ECHO * NUMWAR = number of warehouses *
@ECHO * BLDOPT = full, builddb, objects, objectsfull, bulkload, *
@ECHO * bulkloadfull, or backup *
@ECHO * VERSION = mssql65 or mssql70 *
@ECHO * DBTYPE = normal or scaled *
@ECHO *
@ECHO * Note #1: the BLDOPT and VERSION parameters are case sensitive. *
@ECHO *
@ECHO * Note #2: the DBTYPE is optional. If no DBTYPE is specified, SETUP *
@ECHO * will default to NORMAL. *
@ECHO *
@ECHO * Example: *
@ECHO *
@ECHO * The following command would be used to build a complete 200 *
@ECHO * warehouse database on SQL Server 7.0 running on server \\myserver. *
@ECHO *
@ECHO * SETUP myserver 200 full mssql70 *
@ECHO *
@ECHO * Note, this command file does a backup of the database by default *
@ECHO * after the database build process is complete. If you do not wish *
@ECHO * to make a backup (strongly discouraged), you must edit this file *
@ECHO * and comment that section out. Also, if you need to run the dbcheck *
@ECHO * and the dbtables scripts on the fresh database load for an audit, *
@ECHO * you must either run them manually or edit this file to include them. *
@ECHO *
@ECHO * *****

:end

-- File: CREATEDB.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.22
-- Copyright Microsoft, 2001
-- Purpose: Creates tpcc database and backup files

use master
go

-- Create temporary table for timing

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

```
>
go

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

insert into tpcc_timer values (0,0)
go

-- Store starting time

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

-- create main database files

CREATE DATABASE tpcc
ON PRIMARY
(
    NAME = MSSQL_tpcc_root,
    FILENAME = "C:\MSSQL_tpcc_root.mdf",
    SIZE = 8MB,
    FILEGROWTH = 0),
FILEGROUP MSSQL_misc_fg
(
    NAME = MSSQL_misc1,
    FILENAME = "K:",
    SIZE = 30000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL_misc2,
    FILENAME = "S:",
    SIZE = 30000MB,
    FILEGROWTH = 0),
FILEGROUP MSSQL_cs_fg
(
    NAME = MSSQL_cs1,
    FILENAME = "Y:",
    SIZE = 50000MB,
    FILEGROWTH = 0),
(
    NAME = MSSQL_cs2,
    FILENAME = "W:",
    SIZE = 50000MB,
    FILEGROWTH = 0)
LOG ON
(
    NAME =MSSQL_tpcc_log,
    FILENAME ="L:",
    SIZE =54000MB,
    FILEGROWTH =0)

COLLATE Latin1_General_BIN
go

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

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

-- remove temporary table
```

Appendix B - Database Design

```
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. 4.00
-- Copyright Microsoft, 1996
-- Purpose: Creates TPC-C tables
```

```
use tpcc
go
```

```
if exists ( select name from sysobjects where name = 'warehouse' )
drop table warehouse
```

```
go
create table warehouse
```

```
(
    w_id                smallint,
    w_name              char(10),
    w_street_1         char(20),
    w_street_2         char(20),
    w_city             char(20),
    w_state            char(2),
    w_zip              char(9),
    w_tax              numeric(4,4),
    w_ytd              numeric(12,2)
) on MSSQL70_misc_fg
go
```

```
if exists ( select name from sysobjects where name = 'district' )
drop table district
```

```
go
create table district
```

```
(
    d_id                tinyint,
    d_w_id             smallint,
    d_name             char(10),
    d_street_1        char(20),
    d_street_2        char(20),
    d_city            char(20),
    d_state           char(2),
    d_zip             char(9),
    d_tax             numeric(4,4),
    d_ytd             numeric(12,2),
    d_next_o_id       int
) on MSSQL70_misc_fg
go
```

```
if exists ( select name from sysobjects where name = 'customer' )
drop table customer
```

```
go
create table customer
```

```
(
    c_id                int,
    c_d_id             tinyint,
    c_w_id            smallint,
    c_first            char(16),
```

```
    c_middle          char(2),
    c_last            char(16),
    c_street_1       char(20),
    c_street_2       char(20),
    c_city           char(20),
    c_state          char(2),
    c_zip            char(9),
    c_phone          char(16),
    c_since          datetime,
    c_credit         char(2),
    c_credit_lim     numeric(12,2),
    c_discount       numeric(4,4),
    c_balance        numeric(12,2),
    c_ytd_payment    numeric(12,2),
    c_payment_cnt    smallint,
    c_delivery_cnt   smallint,
    c_data           char(500)
) on MSSQL70_cs_fg
go
```

```
if exists ( select name from sysobjects where name = 'history' )
drop table history
```

```
go
create table history
```

```
(
    h_c_id            int,
    h_c_d_id         tinyint,
    h_c_w_id         smallint,
    h_d_id           tinyint,
    h_w_id           smallint,
    h_date           datetime,
    h_amount         numeric(6,2),
    h_data           char(24)
) on MSSQL70_misc_fg
go
```

```
if exists ( select name from sysobjects where name = 'new_order' )
drop table new_order
```

```
go
create table new_order
```

```
(
    no_o_id          int,
    no_d_id          tinyint,
    no_w_id          smallint
) on MSSQL70_misc_fg
go
```

```
if exists ( select name from sysobjects where name = 'orders' )
drop table orders
```

```
go
create table orders
```

```
(
    o_id            int,
    o_d_id         tinyint,
    o_w_id         smallint,
    o_c_id         int,
    o_entry_d      datetime,
    o_carrier_id   tinyint,
    o_ol_cnt       tinyint,
    o_all_local    tinyint
) on MSSQL70_misc_fg
go
```

Appendix B - Database Design

```
if exists ( select name from sysobjects where name = 'order_line' )
    drop table order_line
go
create table order_line
(
    ol_o_id                int,
    ol_d_id                tinyint,
    ol_w_id                smallint,
    ol_number              tinyint,
    ol_i_id                int,
    ol_supply_w_id         smallint,
    ol_delivery_d          datetime,
    ol_quantity            smallint,
    ol_amount              numeric(6,2),
    ol_dist_info           char(24)
) on MSSQL70_misc_fg
go

if exists ( select name from sysobjects where name = 'item' )
    drop table item
go
create table item
(
    i_id                  int,
    i_im_id               int,
    i_name                char(24),
    i_price                numeric(5,2),
    i_data                char(50)
) on MSSQL70_misc_fg
go

if exists ( select name from sysobjects where name = 'stock' )
    drop table stock
go
create table stock
(
    s_i_id                int,
    s_w_id                smallint,
    s_quantity            smallint,
    s_dist_01             char(24),
    s_dist_02             char(24),
    s_dist_03             char(24),
    s_dist_04             char(24),
    s_dist_05             char(24),
    s_dist_06             char(24),
    s_dist_07             char(24),
    s_dist_08             char(24),
    s_dist_09             char(24),
    s_dist_10            char(24),
    s_ytd                 int,
    s_order_cnt           smallint,
    s_remote_cnt          smallint,
    s_data                char(50)
) on MSSQL70_cs_fg
go
```

idxcuscl.sql

```
-- File:      IDXCUSCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on customer table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'customer_c1' )
    drop index customer.customer_c1

create unique clustered index customer_c1 on customer(c_w_id, c_d_id, c_id)
on MSSQL70_cs_fg

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

go
```

idxcusnc.sql

```
-- File:      IDXCUSNC.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates non-clustered index on customer table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'customer_nc1' )
    drop index customer.customer_nc1

create unique nonclustered index customer_nc1 on customer(c_w_id, c_d_id, c_last,
c_first, c_id)
on MSSQL70_cs_fg

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

go
```


Appendix B - Database Design

idxdiscl.sql

```
-- File:      IDXDISCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on district table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'district_c1' )
    drop index district.district_c1

create unique clustered index  district_c1 on district(d_w_id, d_id)
    with fillfactor=100 on MSSQL70_misc_fg

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

go
```

idxitmcl.sql

```
-- File:      IDXITMCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on item table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'item_c1' )
    drop index item.item_c1

create unique clustered index item_c1 on item(i_id)
    on MSSQL70_misc_fg

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

go
```

idxnodcl.sql

```
-- File:      IDXNODCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on new_order table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'new_order_c1' )
    drop index new_order.new_order_c1

create unique clustered index new_order_c1 on new_order(no_w_id, no_d_id, no_o_id)
    on MSSQL70_misc_fg

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

go
```

idxodcl.sql

```
-- File:      IDXNODCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on new_order table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'new_order_c1' )
    drop index new_order.new_order_c1

create unique clustered index new_order_c1 on new_order(no_w_id, no_d_id, no_o_id)
    on MSSQL70_misc_fg

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

go
```

Appendix B - Database Design

idxordcl.sql

```
-- File:      IDXORDCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on orders table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'orders_c1' )
    drop index orders.orders_c1

create unique clustered index orders_c1 on orders(o_w_id, o_d_id, o_id)
    on MSSQL70_misc_fg

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

go
```

idxstkcl.sql

```
-- File:      IDXSTKCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on stock table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'stock_c1' )
    drop index stock.stock_c1

create unique clustered index stock_c1 on stock(s_i_id, s_w_id)
    on MSSQL70_cs_fg

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

go
```

idxwarcl.sql

```
-- File:      IDXWARCL.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates clustered index on warehouse table

use tpcc
go

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

if exists ( select name from sysindexes where name = 'warehouse_c1' )
    drop index warehouse.warehouse_c1

create unique clustered index warehouse_c1 on warehouse(w_id)
    with fillfactor=100 on MSSQL70_misc_fg

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

go
```

dbopt1.sql

```
-- File:      DBOPT1.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Sets database options for data load

use master
go

exec sp_dboption tpcc,'select into/bulkcopy',true
exec sp_dboption tpcc,'trunc. log on chkpt.',true
go

use tpcc
go

checkpoint
go
```

Appendix B - Database Design

dbopt2.sql

```
-- File:      DBOPT2.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Resets database options after data load

use master
go

sp_dboption tpcc,'select ',false
go

sp_dboption tpcc,'trunc. ',false
go

use tpcc
go

checkpoint
go

sp_configure allow,1
go

reconfigure with override
go

/*
/* Set option values for user-defined indexes */
/*
/*

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

Print ' '
Print '*****'
Print 'Pre-specified Locking Hierarchy:'
Print '  Lockflag = 0 ==> No pre-pecified hierarchy'
Print '  Lockflag = 1 ==> Lock at Page-level then Table-level'
Print '  Lockflag = 2 ==> Lock at Row-level then Table-level'
Print '  Lockflag = 3 ==> Lock at Table-level'
Print ' '

select name,lockflags
```

```
from sysindexes
where object_id("warehouse")=id or
      object_id("district")=id or
      object_id("customer")=id or
      object_id("stock")=id or
      object_id("orders")=id or
      object_id("order_line")=id or
      object_id("history")=id or
      object_id("new_order")=id or
      object_id("item")=id

order by lockflags asc
go

sp_configure allow,0
go

reconfigure with override
go

exec sp_dboption tpcc, 'auto update statistics', FALSE
exec sp_dboption tpcc, 'auto create statistics', FALSE
go

exec sp_tableoption "district","pintable",true
exec sp_tableoption "warehouse","pintable",true
exec sp_tableoption "new_order","pintable",true
exec sp_tableoption "item","pintable",true
go
```

dbopt3.sql

```
use tpcc
go
sp_indexoption 'orders','AllowPageLocks',TRUE
go
sp_indexoption 'orders','AllowRowLocks',FALSE
go
sp_indexoption 'order_line','AllowPageLocks',TRUE
go
sp_indexoption 'order_line','AllowRowLocks',FALSE
go
```

backup.sql

Appendix B - Database Design

```
-- File:      BACKUP.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.22
--           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)

backup 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. 4.22
--           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
```

Appendix B - Database Design

Stored Procedures

neword.sql

```
-- File:      NEWORD.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.01
--           Copyright Microsoft, 1996
-- Purpose:   Creates new order transaction stored procedure
--
-- Modified 9/21/98 - Jamie Reding - Microsoft Corporation
--               Reordered @rowcount check so that invalid supply warehouse id,
--               as well as invalid item id, is detected and causes explicit
--               transaction rollback.
--
use tpcc
go

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

create proc tpcc_neworder

    @w_id          smallint,
    @d_id          tinyint,
    @c_id          int,
    @o_ol_cnt      tinyint,
    @o_all_local   tinyint,
    @i_id1 int = 0, @s_w_id1 smallint = 0,
    @i_id2 int = 0, @s_w_id2 smallint = 0,
    @i_id3 int = 0, @s_w_id3 smallint = 0,
    @i_id4 int = 0, @s_w_id4 smallint = 0,
    @i_id5 int = 0, @s_w_id5 smallint = 0,
    @i_id6 int = 0, @s_w_id6 smallint = 0,
    @i_id7 int = 0, @s_w_id7 smallint = 0,
    @i_id8 int = 0, @s_w_id8 smallint = 0,
    @i_id9 int = 0, @s_w_id9 smallint = 0,
    @i_id10 int = 0, @s_w_id10 smallint =
    @i_id11 int = 0, @s_w_id11 smallint =
    @i_id12 int = 0, @s_w_id12 smallint =
    @i_id13 int = 0, @s_w_id13 smallint =

    @ol_qty1 smallint = 0,
    @ol_qty2 smallint = 0,
    @ol_qty3 smallint = 0,
    @ol_qty4 smallint = 0,
    @ol_qty5 smallint = 0,
    @ol_qty6 smallint = 0,
    @ol_qty7 smallint = 0,
    @ol_qty8 smallint = 0,
    @ol_qty9 smallint = 0,
    @ol_qty10 smallint = 0,
    @ol_qty11 smallint = 0,
    @ol_qty12 smallint = 0,
    @ol_qty13 smallint = 0,
```

```
    @i_id14 int = 0, @s_w_id14 smallint =
    @i_id15 int = 0, @s_w_id15 smallint =

    @ol_qty14 smallint = 0,
    @ol_qty15 smallint = 0

as
declare  @w_tax          numeric(4,4),
         @d_tax          numeric(4,4),
         @c_last         char(16),
         @c_credit       char(2),
         @c_discount     numeric(4,4),
         @i_price        numeric(5,2),
         @i_name         char(24),
         @i_data         char(50),
         @o_entry_d      datetime,
         @remote_flag    int,
         @s_quantity     smallint,
         @s_data         char(50),
         @s_dist         char(24),
         @li_no          int,
         @o_id           int,
         @commit_flag    tinyint,
         @li_id          int,
         @li_s_w_id      smallint,
         @li_qty         smallint,
         @ol_number      int,
         @c_id_local     int

begin

    begin transaction n

    -- get district tax and next available order id and update
    -- plus initialize local variables

        update  district
        set     @d_tax          = d_tax,
               @o_id           = d_next_o_id,
               d_next_o_id    = d_next_o_id + 1,
               @o_entry_d      = getdate(),
               @li_no          = 0,
               @commit_flag    = 1

        where   d_w_id          = @w_id and
               d_id            = @d_id

    -- process orderlines

    while (@li_no < @o_ol_cnt)
        begin

            select @li_no = @li_no + 1

    -- set i_id, s_w_id, and qty for this lineitem

            select @li_id = case @li_no
                when 1 then @i_id1
                when 2 then @i_id2
                when 3 then @i_id3
                when 4 then @i_id4
                when 5 then @i_id5
                when 6 then @i_id6
                when 7 then @i_id7
```

Appendix B - Database Design

```
when 8 then @i_id8
when 9 then @i_id9
when 10 then @i_id10
when 11 then @i_id11
when 12 then @i_id12
when 13 then @i_id13
when 14 then @i_id14
when 15 then @i_id15
end,

@li_s_w_id = case @li_no
when 1 then @s_w_id1
when 2 then @s_w_id2
when 3 then @s_w_id3
when 4 then @s_w_id4
when 5 then @s_w_id5
when 6 then @s_w_id6
when 7 then @s_w_id7
when 8 then @s_w_id8
when 9 then @s_w_id9
when 10 then @s_w_id10
when 11 then @s_w_id11
when 12 then @s_w_id12
when 13 then @s_w_id13
when 14 then @s_w_id14
when 15 then @s_w_id15
end,

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

-- get item data (no one updates item)
select  @i_price = i_price,
        @i_name = i_name,
        @i_data = i_data
from    item (tablock repeatableread)
where   i_id = @li_id

-- update stock values
update  stock
set     s_ytd      = s_ytd + @li_qty,
        @s_quantity = s_quantity - @li_qty +
        case when (s_quantity - @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
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
else
end
```

Appendix B - Database Design

```
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. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates payment transaction stored procedure

use tpcc
go

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

create proc tpcc_payment @w_id          smallint,
                        @c_w_id        smallint,
                        @h_amount      numeric(6,2),
                        @d_id          tinyint,
                        @c_d_id        tinyint,
                        @c_id          int,
                        @c_last        char(16) =
""

as
declare @w_street_1 char(20),
        @w_street_2 char(20),
        @w_city     char(20),
        @w_state    char(2),
        @w_zip      char(9),
        @w_name     char(10),
        @d_street_1 char(20),
        @d_street_2 char(20),
        @d_city     char(20),
        @d_state    char(2),
        @d_zip      char(9),
        @d_name     char(10),
        @c_first    char(16),
        @c_middle   char(2),
        @c_street_1 char(20),
        @c_street_2 char(20),
        @c_city     char(20),
        @c_state    char(2),
        @c_zip      char(9),
        @c_phone    char(16),
        @c_since    datetime,
        @c_credit   char(2),
        @c_credit_lim numeric(12,2),
        @c_balance  numeric(12,2),
        @c_discount numeric(4,4),
        @data       char(500),
        @c_data     char(500),
        @datetime   datetime,
        @w_ytd      numeric(12,2),
```

Appendix B - Database Design

```
@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,
```


Appendix B - Database Design

```
@w_id_local,
@datetime,
@h_amount,
+ " " + @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. 4.00
--         Copyright Microsoft, 1996
-- Purpose: Creates order status transaction stored procedure

use tpcc
go

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

create proc tpcc_orderstatus @w_id          smallint,
```

```
        tinyint,
        int,
= ""
as
declare @c_balance      numeric(12,2),
        @c_first       char(16),
        @c_middle      char(2),
        @c_id          int,
        @o_entry_d     datetime,
        @o_carrier_id  smallint,
        @cnt           smallint
begin tran o
        if (@c_id = 0)
            begin
-- get customer id and info using last name

                select @cnt = (count(*)+1)/2
                    from customer (repeatableread)
                where c_last = @c_last and
                      c_w_id = @w_id and
                      c_d_id = @d_id

                set rowcount @cnt

                select @c_id = c_id,
                       @c_balance = c_balance,
                       @c_first  = c_first,
                       @c_last   = c_last,
                       @c_middle = c_middle
                    from customer (repeatableread)
                where c_last = @c_last and
                      c_w_id = @w_id and
                      c_d_id = @d_id
                order by c_w_id, c_d_id, c_last, c_first

                set rowcount 0
            end

        else
            begin
-- get customer info if by id

                select @c_balance = c_balance,
                       @c_first  = c_first,
                       @c_middle = c_middle,
                       @c_last   = c_last
                    from customer (repeatableread)
                where c_id = @c_id and
                      c_d_id = @d_id and
                      c_w_id = @w_id

                select @cnt = @@rowcount
            end
        end
```

Appendix B - Database Design

```
-- if no such customer
    if (@cnt = 0)
    begin
        raiserror("Customer not found",18,1)
        goto custnotfound
    end
-- get order info
    select @o_id = o_id,
           @o_entry_d = o_entry_d,
           @o_carrier_id = o_carrier_id
    from orders (serializable)
    where o_c_id = @c_id and
          o_d_id = @d_id and
          o_w_id = @w_id
    order by o_id asc
-- select order lines for the current order
    select ol_supply_w_id,
           ol_i_id,
           ol_quantity,
           ol_amount,
           ol_delivery_d
    from order_line (repeatableread)
    where ol_o_id = @o_id and
          ol_d_id = @d_id and
          ol_w_id = @w_id

custnotfound:
commit tran o
-- return data to client
select @c_id,
       @c_last,
       @c_first,
       @c_middle,
       @o_entry_d,
       @o_carrier_id,
       @c_balance,
       @o_id

go
```

delivery.sql

```
-- File:      DELIVERY.SQL
-- Microsoft TPC-C Benchmark Kit Ver. 4.00
-- Copyright Microsoft, 1996
-- Purpose:   Creates delivery transaction stored procedure
```

```
use tpcc
go
if exists (select name from sysobjects where name = "tpcc_delivery" )
    drop procedure tpcc_delivery
go
create proc tpcc_delivery    @w_id            smallint,
                             @o_carrier_id  smallint
as
declare @d_id tinyint,
        @o_id int,
        @c_id int,
        @total numeric(12,2),
        @oid1 int,
        @oid2 int,
        @oid3 int,
        @oid4 int,
        @oid5 int,
        @oid6 int,
        @oid7 int,
        @oid8 int,
        @oid9 int,
        @oid10 int

select @d_id = 0

begin tran d

    while (@d_id < 10)
    begin

        select @d_id = @d_id + 1,
               @total = 0,
               @o_id = 0

                select top 1 @o_id = no_o_id
                from new_order (serializable uplock)
                where no_w_id = @w_id and
                      no_d_id = @d_id
                order by no_o_id asc

        if (@@rowcount <> 0)
        begin
-- claim the order for this district

            delete new_order
            where no_w_id = @w_id and
                  no_d_id = @d_id and
                  no_o_id = @o_id

-- set carrier_id on this order (and get customer id)

            update orders
                set o_carrier_id = @o_carrier_id,
                    @c_id = o_c_id
            where o_w_id = @w_id and
                  o_d_id = @d_id and
                  o_id = @o_id
```

Appendix B - Database Design

```
-- set date in all lineitems for this order (and sum amounts)

update order_line
  set ol_delivery_d = getdate(),
      @total        = @total + ol_amount
where ol_w_id = @w_id and
      ol_d_id = @d_id and
      ol_o_id = @o_id

-- accumulate lineitem amounts for this order into customer

update customer
  set c_balance      = c_balance + @total,
      c_delivery_cnt = c_delivery_cnt + 1
where c_w_id = @w_id and
      c_d_id = @d_id and
      c_id   = @c_id

end

select @oid1 = case @d_id when 1 then @o_id else @oid1 end,
       @oid2 = case @d_id when 2 then @o_id else @oid2 end,
       @oid3 = case @d_id when 3 then @o_id else @oid3 end,
       @oid4 = case @d_id when 4 then @o_id else @oid4 end,
       @oid5 = case @d_id when 5 then @o_id else @oid5 end,
       @oid6 = case @d_id when 6 then @o_id else @oid6 end,
       @oid7 = case @d_id when 7 then @o_id else @oid7 end,
       @oid8 = case @d_id when 8 then @o_id else @oid8 end,
       @oid9 = case @d_id when 9 then @o_id else @oid9 end,
       @oid10 = case @d_id when 10 then @o_id else @oid10 end

end

commit tran d

-- return delivery data to client

select @oid1,
       @oid2,
       @oid3,
       @oid4,
       @oid5,
       @oid6,
       @oid7,
       @oid8,
       @oid9,
       @oid10

go
```

stocklev.sql

```
-- File:      STOCKLEV.SQL
--           Microsoft TPC-C Benchmark Kit Ver. 4.00
--           Copyright Microsoft, 1996
-- Purpose:   Creates stock level transaction stored procedure

use tpcc
go
```

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

create proc tpcc_stocklevel @w_id          smallint,
                           @d_id          tinyint,
                           @threshold    smallint

as

  declare @o_id_low int,
          @o_id_high int

  select @o_id_low = (d_next_o_id - 20),
         @o_id_high = (d_next_o_id - 1)
  from district
  where d_w_id = @w_id and
        d_id   = @d_id

  select count(distinct(s_i_id))
  from stock, order_line
  where ol_w_id = @w_id and
        ol_d_id = @d_id and
        ol_o_id between @o_id_low and @o_id_high and
        s_w_id = ol_w_id and
        s_i_id = ol_i_id and
        s_quantity < @threshold

go
```

Loader Source Code

tpcc.h

```
// File:      TPCC.H
//           Microsoft TPC-C Kit Ver. 4.00
//           Copyright Microsoft, 1996, 1997, 1998

// Purpose:   Header file for TPC-C database loader

// Build number of TPC Benchmark Kit
#define TPCKIT_VER "4.00"

// General headers
#include <windows.h>
#include <winbase.h>
#include <stdlib.h>
#include <stdio.h>
#include <process.h>
#include <stddef.h>
#include <stdarg.h>
#include <string.h>
```

Appendix B - Database Design

```

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

// ODBC headers
#include <sql.h>
#include <sqlext.h>
#include <odbcss.h>

// General constants
#define MILLI 1000
#define FALSE 0
#define TRUE 1
#define UNDEF -1
#define MINPRINTASCII 32
#define MAXPRINTASCII 126

// Default environment constants
#define SERVER ""
#define DATABASE "tpcc"
#define USER "sa"
#define PASSWORD ""

// Default loader arguments
#define BATCH 10000
#define DEFILDPACKSIZE 32768
#define ORDERS_PER_DIST 3000
#define LOADER_RES_FILE "logs\\load.out"
#define LOADER_NURAND_C 123
#define DEF_STARTING_WAREHOUSE 1
#define BUILD_INDEX 1 // build both data
and indexes
#define INDEX_ORDER 1 // build indexes
before load
#define SCALE_DOWN 0 // build a normal scale
database
#define INDEX_SCRIPT_PATH "scripts"

typedef struct
{
    char *server;
    char *database;
    char *user;
    char *password;
    BOOL tables_all;
    // set if loading all tables
    BOOL table_item;
    // set if loading ITEM table specifically
    BOOL table_warehouse; // set if loading
WAREHOUSE, DISTRICT, and STOCK
    BOOL table_customer; // set if
loading CUSTOMER and HISTORY
    BOOL table_orders; // set if
loading NEW-ORDER, ORDERS, ORDER-LINE
    long num_warehouses;
    long batch;
    long verbose;
    long pack_size;
    char *loader_res_file;
    char *synch_servername;
    long case_sensitivity;
    long starting_warehouse;
    long build_index;

```

```

long index_order;
long scale_down;
char *index_script_path;
} TPCCCLR_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();

```

Appendix B - Database Design

```
void PaddString();
```

tpccldr.c

```
// File: TPCCLDR.C
// Microsoft TPC-C Kit Ver. 4.00
// Copyright Microsoft, 1996, 1997, 1998
// Purpose: Source file for TPC-C database loader

// Includes
#include "tpcc.h"
#include "search.h"

// Defines
#define MAXITEMS 100000
#define MAXITEMS_SCALE_DOWN 100
#define CUSTOMERS_PER_DISTRICT 3000
#define CUSTOMERS_SCALE_DOWN 30
#define DISTRICT_PER_WAREHOUSE 10
#define ORDERS_PER_DISTRICT 3000
#define ORDERS_SCALE_DOWN 30
#define MAX_CUSTOMER_THREADS 2
#define MAX_ORDER_THREADS 3
#define MAX_MAIN_THREADS 4

// Functions declarations

void HandleErrorDBC (SQLHDBC hdbc1);

long NURand();
void LoadItem();
void LoadWarehouse();

void Stock();
void District();

void LoadCustomer();
void CustomerBufInit();
void CustomerBufLoad();
void LoadCustomerTable();
void LoadHistoryTable();

void LoadOrders();
void OrdersBufInit();
void OrdersBufLoad();
void LoadOrdersTable();
void LoadNewOrderTable();
void LoadOrderLineTable();
void GetPermutation();
void CheckForCommit();
void OpenConnections();
void BuildIndex();
void FormatDate ();
```

```
// Shared memory structures

typedef struct
{
    long ol;
    long ol_i_id;
    short ol_supply_w_id;
    short ol_quantity;
    double ol_amount;
    char ol_dist_info[DIST_INFO_LEN+1];
    char ol_delivery_d[OL_DELIVERY_D_LEN+1];
} ORDER_LINE_STRUCT;

typedef struct
{
    long o_id;
    short o_d_id;
    short o_w_id;
    long o_c_id;
    short o_carrier_id;
    short o_ol_cnt;
    short o_all_local;
    ORDER_LINE_STRUCT o_ol[15];
} ORDERS_STRUCT;

typedef struct
{
    long c_id;
    short c_d_id;
    short c_w_id;
    char c_first[FIRST_NAME_LEN+1];
    char c_middle[MIDDLE_NAME_LEN+1];
    char c_last[LAST_NAME_LEN+1];
    char c_street_1[ADDRESS_LEN+1];
    char c_street_2[ADDRESS_LEN+1];
    char c_city[ADDRESS_LEN+1];
    char c_state[STATE_LEN+1];
    char c_zip[ZIP_LEN+1];
    char c_phone[PHONE_LEN+1];
    char c_credit[ CREDIT_LEN+1];
    double c_credit_lim;
    double c_discount;
    // fix to avoid ODBC float to numeric conversion problem.
    // double c_balance;
    char c_balance[6];

    double c_ytd_payment;
    short c_payment_cnt;
    short c_delivery_cnt;
    char c_data[C_DATA_LEN+1];
    double h_amount;
    char h_data[H_DATA_LEN+1];
} CUSTOMER_STRUCT;

typedef struct
{
    char c_last[LAST_NAME_LEN+1];
    char c_first[FIRST_NAME_LEN+1];
    long c_id;
} CUSTOMER_SORT_STRUCT;

typedef struct
{
```

Appendix B - Database Design

```
    long          time_start;
} LOADER_TIME_STRUCT;

// Global variables

char          szLastError[300];

HENV          henv;

HDBC          i_hdbc1;          // for ITEM table
HDBC          w_hdbc1;          // for WAREHOUSE, DISTRICT,
STOCK
HDBC          c_hdbc1;          // for CUSTOMER
HDBC          c_hdbc2;          // for HISTORY
HDBC          o_hdbc1;          // for ORDERS
HDBC          o_hdbc2;          // for NEW-ORDER

HDBC          o_hdbc3;          // for ORDER-LINE

HSTMT         i_hstmt1;
HSTMT         w_hstmt1;
HSTMT         c_hstmt1, c_hstmt2;
HSTMT         o_hstmt1, o_hstmt2, o_hstmt3;

ORDERS_STRUCT orders_buf[ORDERS_PER_DISTRICT];
CUSTOMER_STRUCT customer_buf[CUSTOMERS_PER_DISTRICT];
long          orders_rows_loaded;
long          new_order_rows_loaded;
long          order_line_rows_loaded;
long          history_rows_loaded;
long          customer_rows_loaded;
long          stock_rows_loaded;
long          district_rows_loaded;
long          item_rows_loaded;
long          warehouse_rows_loaded;
long          main_time_start;
long          main_time_end;
long          max_items;
long          customers_per_district;
long          orders_per_district;
long          first_new_order;
long          last_new_order;

TPCCLDR_ARGS  *aptr, args;

//=====
//
// Function name: main
//
//=====

int main(int argc, char **argv)
{
    DWORD          dwThreadID[MAX_MAIN_THREADS];
    HANDLE          hThread[MAX_MAIN_THREADS];
    FILE           *fLoader;
    char           buffer[255];
    int            i;
```

```
    for (i=0; i<MAX_MAIN_THREADS; i++)
        hThread[i] = NULL;

    printf("\n*****");
    printf("\n*");
    printf("\n* Microsoft SQL Server");
    printf("\n*");
    printf("\n* TPC-C BENCHMARK KIT: Database loader");
    printf("\n* Version %s", TPCKIT_VER);
    printf("\n*");
    printf("\n*****\n\n");

    // process command line arguments

    aptr = &args;
    GetArgsLoader(argc, argv, aptr);

    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
```

Appendix B - Database Design

```
    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");
                }
            }
            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");
                }
                // 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
    //
    //=====
```

Appendix B - Database Design

```
void LoadItem()
{
    long          i_id;
    long          i_im_id;
    char          i_name[I_NAME_LEN+1];
    double        i_price;
    char          i_data[I_DATA_LEN+1];
    char          name[20];
    long          time_start;
    RETCODE       rc;
    DBINT         rcint;
    char          bcphint[128];

    // Seed with unique number
    seed(1);

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

    // if build index before load
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
        BuildIndex("idxitmc1");

    InitString(i_name, I_NAME_LEN+1);
    InitString(i_data, I_DATA_LEN+1);

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

    rc = bcp_init(i_hdbc1, name, NULL, "logs\\item.err", DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(i_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (i_id), ROWS_PER_BATCH = 100000");
        rc = bcp_control(i_hdbc1, BCPHINTS, (void*) bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(i_hdbc1);
    }

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

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

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

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

    rc = bcp_bind(i_hdbc1, (BYTE *) i_data, 0, I_DATA_LEN, NULL, 0, 0, 5);
    if (rc != SUCCEEDED)
        HandleErrorDBC(i_hdbc1);
}
```

```
time_start = (TimeNow() / MILLI);

item_rows_loaded = 0;

for (i_id = 1; i_id <= max_items; i_id++)
{
    i_im_id = RandomNumber(1L, 10000L);

    MakeAlphaString(14, 24, I_NAME_LEN, i_name);

    i_price = ((float) RandomNumber(100L, 10000L))/100.0;

    MakeOriginalAlphaString(26, 50, I_DATA_LEN, i_data, 10);

    rc = bcp_sendrow(i_hdbc1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(i_hdbc1);

    item_rows_loaded++;
    CheckForCommit(i_hdbc1, i_hstmt1, item_rows_loaded, "item",
&time_start);
}

rcint = bcp_done(i_hdbc1);
if (rcint < 0)
    HandleErrorDBC(i_hdbc1);

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

SQLFreeStmt(i_hstmt1, SQL_DROP);
SQLDisconnect(i_hdbc1);
SQLFreeConnect(i_hdbc1);

// if build index after load
if ((aptr->build_index == 1) && (aptr->index_order == 0))
    BuildIndex("idxitmc1");
}

//=====
//
// 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;
}
```


Appendix B - Database Design

```
char    bcphint[128];

// Seed with unique number
seed(2);

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

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

InitString(w_name, W_NAME_LEN+1);
InitAddress(w_street_1, w_street_2, w_city, w_state, w_zip);

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

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

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

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

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

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

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

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

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

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

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

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

time_start = (TimeNow() / MILLI);

warehouse_rows_loaded = 0;

for (w_id = (short)aptr->starting_warehouse; w_id <= aptr->num_warehouses;
w_id++)
{
    MakeAlphaString(6,10, W_NAME_LEN, w_name);

    MakeAddress(w_street_1, w_street_2, w_city, w_state, w_zip);

    w_tax = ((float) RandomNumber(0L,2000L))/10000.00;

    w_ytd = 300000.00;

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

    warehouse_rows_loaded++;
    CheckForCommit(w_hdbc1, i_hstmt1, warehouse_rows_loaded, "warehouse",
&time_start);
}

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

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

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

stock_rows_loaded = 0;
district_rows_loaded = 0;

District();
Stock();

}

//=====
//
// Function : District
//
//=====

void District()
{
    short d_id;
    short d_w_id;
    char d_name[D_NAME_LEN+1];
    char d_street_1[ADDRESS_LEN+1];
    char d_street_2[ADDRESS_LEN+1];
    char d_city[ADDRESS_LEN+1];
```

Appendix B - Database Design

```
char d_state[STATE_LEN+1];
char d_zip[ZIP_LEN+1];
double d_tax;
double d_ytd;
char name[20];
long d_next_o_id;
long time_start;
int w_id;
RETCODE rc;
DBINT rcint;
char bcphint[128];

// Seed with unique number
seed(4);

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

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

InitString(d_name, D_NAME_LEN+1);
InitAddress(d_street_1, d_street_2, d_city, d_state, d_zip);
sprintf(name, "%s..%s", aptr->database, "district");

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

if ((aptr->build_index == 1) && (aptr->index_order == 1))
{
    sprintf(bcphint, "tablock, order (d_w_id, d_id), ROWS_PER_BATCH =
%u", (aptr->num_warehouses * 10));
    rc = bcp_control(w_hdbc1, BCPHINTS, (void*) bcphint);
    if (rc != SUCCEEDED)
        HandleErrorDBC(w_hdbc1);
}

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

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

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

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

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

d_ytd = 30000.0;
d_next_o_id = orders_per_district+1;
time_start = (TimeNow() / MILLI);

for (w_id = aptr->starting_warehouse; w_id <= aptr->num_warehouses; w_id++)
{
    d_w_id = w_id;
    for (d_id = 1; d_id <= DISTRICT_PER_WAREHOUSE; d_id++)
    {
        MakeAlphaString(6,10,D_NAME_LEN, d_name);
        MakeAddress(d_street_1, d_street_2, d_city, d_state,
d_zip);
        d_tax = ((float) RandomNumber(0L,2000L))/10000.00;
        rc = bcp_sendrow(w_hdbc1);
        if (rc != SUCCEEDED)
            HandleErrorDBC(w_hdbc1);
        district_rows_loaded++;
        CheckForCommit(w_hdbc1, w_hstmt1, district_rows_loaded,
"district", &time_start);
    }
}

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

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

// if build index after load...
if ((aptr->build_index == 1) && (aptr->index_order == 0))
    BuildIndex("idxdiscl");
```

Appendix B - Database Design

```
    return;
}

//=====
//
// Function   : Stock
//
//=====

void Stock()
{
    long   s_i_id;
    short  s_w_id;
    short  s_quantity;
    char   s_dist_01[S_DIST_LEN+1];
    char   s_dist_02[S_DIST_LEN+1];
    char   s_dist_03[S_DIST_LEN+1];
    char   s_dist_04[S_DIST_LEN+1];
    char   s_dist_05[S_DIST_LEN+1];
    char   s_dist_06[S_DIST_LEN+1];
    char   s_dist_07[S_DIST_LEN+1];
    char   s_dist_08[S_DIST_LEN+1];
    char   s_dist_09[S_DIST_LEN+1];
    char   s_dist_10[S_DIST_LEN+1];
    long   s_ytd;
    short  s_order_cnt;
    short  s_remote_cnt;
    char   s_data[S_DATA_LEN+1];
    short  len;
    char   name[20];
    long   time_start;
    RETCODE rc;
    DBINT  rcint;
    char   bcphint[128];

    // Seed with unique number
    seed(3);

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

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

    rc = bcp_init(w_hdbc1, name, NULL, "logs\\stock.err", DB_IN);
    if (rc != 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 * 100000));
        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);
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Appendix B - Database Design

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

s_ytd = s_order_cnt = s_remote_cnt = 0;

time_start = (TimeNow() / MILLI);

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

for (s_i_id=1; s_i_id <= max_items; s_i_id++)
{
    for (s_w_id = (short)aptr->starting_warehouse; s_w_id <= aptr-
>num_warehouses; s_w_id++)
    {
        s_quantity = (short)RandomNumber(10L,100L);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_01);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_02);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_03);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_04);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_05);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_06);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_07);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_08);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_09);
        len = MakeAlphaString(24,24,S_DIST_LEN, s_dist_10);

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

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

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

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

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

SQLFreeStmt(w_hstmt1, SQL_DROP);
SQLDisconnect(w_hdbc1);
SQLFreeConnect(w_hdbc1);

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

return;
}

//=====
```

```
//
// Function : LoadCustomer
//
//=====

void LoadCustomer()
{
    LOADER_TIME_STRUCT customer_time_start;
    LOADER_TIME_STRUCT history_time_start;
    short w_id;

    short d_id;

    DWORD dwThreadID[MAX_CUSTOMER_THREADS];
    HANDLE hThread[MAX_CUSTOMER_THREADS];
    char name[20];

    RETCODE rc;
    DBINT rcint;
    char bcphint[128];
    char cmd[256];
    char rc_l;
    // SQLRETURN // SQLSMALLINT // SQLCHAR // SQLINTEGER
    Msg[SQL_MAX_MESSAGE_LENGTH]; // 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("idxxcuscl");

    // Initialize bulk copy
    sprintf(name, "%s.%s", aptr->database, "customer");

    rc = bcp_init(c_hdbc1, name, NULL, "logs\\customer.err", DB_IN);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (c_w_id, c_d_id, c_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 30000));
        rc = bcp_control(c_hdbc1, BCPHINTS, (void*) bcphint);
        if (rc != SUCCEED)
            HandleErrorDBC(c_hdbc1);
    }

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

    rc = bcp_init(c_hdbc2, name, NULL, "logs\\history.err", DB_IN);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    sprintf(bcphint, "tablock");
    rc = bcp_control(c_hdbc2, BCPHINTS, (void*) bcphint);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    customer_rows_loaded = 0;
    history_rows_loaded = 0;
}
```

Appendix B - Database Design

```
CustomerBufInit();

customer_time_start.time_start = (TimeNow() / MILLI);
history_time_start.time_start = (TimeNow() / MILLI);

for (w_id = (short)aptr->starting_warehouse; w_id <= aptr->num_warehouses;
w_id++)
{
    for (d_id = 1; d_id <= DISTRICT_PER_WAREHOUSE; d_id++)
    {
        CustomerBufLoad(d_id, w_id);
        // Start parallel loading threads here...
        // Start customer table thread
        printf("...Loading customer table for: d_id = %d, w_id =
%d\n", d_id, w_id);

        hThread[0] = CreateThread(NULL,

0,
(LPTHREAD_START_ROUTINE) LoadCustomerTable,
&customer_time_start,
0,
&dwThreadID[0]);

        if (hThread[0] == NULL)
        {
            printf("Error, failed in creating creating thread
= 0.\n");
            exit(-1);
        }
        // Start History table thread
        printf("...Loading history table for: d_id = %d, w_id =
%d\n", d_id, w_id);

        hThread[1] = CreateThread(NULL,

0,
(LPTHREAD_START_ROUTINE) LoadHistoryTable,
&history_time_start,
0,
&dwThreadID[1]);

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

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

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

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

}

// flush the bulk connection
rcint = bcp_done(c_hdbc1);
if (rcint < 0)
    HandleErrorDBC(c_hdbc1);

rcint = bcp_done(c_hdbc2);
if (rcint < 0)
    HandleErrorDBC(c_hdbc2);

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

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

// build non-clustered index
if (aptr->build_index == 1)
    BuildIndex("idxcusnc");

// Output the NURAND used for the loader into C_FIRST for C_ID = 1,
// C_W_ID = 1, and C_D_ID = 1
sprintf(cmd, "isql -S%s -U%s -P%s -d%s -e -Q\"update customer set c_first =
'C_LOAD = %d' where c_id = 1 and c_w_id = 1 and c_d_id = 1\" > logs\\nurand_load.log",
aptr->server,
aptr->user,
aptr->password,
aptr->database,
LOADER_NURAND_C);

system(cmd);

SQLFreeStmt(c_hstmt1, SQL_DROP);
SQLDisconnect(c_hdbc1);
SQLFreeConnect(c_hdbc1);

SQLFreeStmt(c_hstmt2, SQL_DROP);
SQLDisconnect(c_hdbc2);
SQLFreeConnect(c_hdbc2);

return;
}
```

Appendix B - Database Design

```
//=====
//
// 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");
```

Appendix B - Database Design

```
        MakeAlphaString(500, 500, C_DATA_LEN, customer_buf[i].c_data);

        // Generate HISTORY data
        MakeAlphaString(12, 24, H_DATA_LEN, customer_buf[i].h_data);
    }

//=====
//
// Function   : LoadCustomerTable
//
//=====

void LoadCustomerTable(LOADER_TIME_STRUCT *customer_time_start)
{
    int         i;
    long        c_id;
    short       c_d_id;
    short       c_w_id;
    char        c_first[FIRST_NAME_LEN+1];
    char        c_middle[MIDDLE_NAME_LEN+1];
    char        c_last[LAST_NAME_LEN+1];
    char        c_street_1[ADDRESS_LEN+1];
    char        c_street_2[ADDRESS_LEN+1];
    char        c_city[ADDRESS_LEN+1];
    char        c_state[STATE_LEN+1];
    char        c_zip[ZIP_LEN+1];
    char        c_phone[PHONE_LEN+1];
    char        c_credit[CREDIT_LEN+1];
    double      c_credit_lim;
    double      c_discount;

    // fix to avoid ODBC float to numeric conversion problem.
    // double      c_balance;
    char        c_balance[6];

    double      c_ytd_payment;
    short       c_payment_cnt;
    short       c_delivery_cnt;
    char        c_data[C_DATA_LEN+1];
    char        c_since[C_SINCE_LEN+1];
    RETCODE     rc;

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

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

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

    rc = bcp_bind(c_hdbc1, (BYTE *) c_first, 0, FIRST_NAME_LEN, NULL, 0, 0, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) c_middle, 0, MIDDLE_NAME_LEN, NULL, 0, 0, 5);
```

```
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

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

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

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

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

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

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

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

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

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

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

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_discount, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8,
16);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

    // fix to avoid ODBC float to numeric conversion problem.
    // rc = bcp_bind(c_hdbc1, (BYTE *) &c_balance, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8,
17);
    // if (rc != SUCCEEDED)
    //     HandleErrorDBC(c_hdbc1);
    rc = bcp_bind(c_hdbc1, (BYTE *) c_balance, 0, 5, NULL, 0, SQLCHARACTER, 17);
    if (rc != SUCCEEDED)
        HandleErrorDBC(c_hdbc1);

    rc = bcp_bind(c_hdbc1, (BYTE *) &c_ytd_payment, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8,
18);
```

Appendix B - Database Design

```
        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)
```


Appendix B - Database Design

```
        HandleErrorDBC(c_hdbc2);

        history_rows_loaded++;
        CheckForCommit(c_hdbc2, c_hstmt2, history_rows_loaded, "history",
&history_time_start->time_start);
    }
}

//=====
//
// Function   : LoadOrders
//
//=====

void LoadOrders()
{
    LOADER_TIME_STRUCT    orders_time_start;
    LOADER_TIME_STRUCT    new_order_time_start;
    LOADER_TIME_STRUCT    order_line_time_start;
    short                 w_id;

    short                 d_id;

    DWORD                 dwThreadID[MAX_ORDER_THREADS];
    HANDLE                 hThread[MAX_ORDER_THREADS];
    char                   name[20];

    RETCODE                rc;
    char                   bcphint[128];

    // seed with unique number
    seed(6);

    printf("Loading orders...\n");

    // if build index before load...
    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        BuildIndex("idxordcl");
        BuildIndex("idxnodcl");
        BuildIndex("idxodlcl");
    }

    // initialize bulk copy
    sprintf(name, "%s..%s", aptr->database, "orders");

    rc = bcp_init(o_hdbc1, name, NULL, "logs\\orders.err", DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    if ((aptr->build_index == 1) && (aptr->index_order == 1))
    {
        sprintf(bcphint, "tablock, order (o_w_id, o_d_id, o_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 30000));
        rc = bcp_control(o_hdbc1, BCPHINTS, (void*) bcphint);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc1);
    }

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

    rc = bcp_init(o_hdbc2, name, NULL, "logs\\neword.err", DB_IN);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc2);
```

```
        if ((aptr->build_index == 1) && (aptr->index_order == 1))
        {
            sprintf(bcphint, "tablock, order (no_w_id, no_d_id, no_o_id),
ROWS_PER_BATCH = %u", (aptr->num_warehouses * 9000));
            rc = bcp_control(o_hdbc2, BCPHINTS, (void*) bcphint);
            if (rc != SUCCEEDED)
                HandleErrorDBC(o_hdbc2);
        }

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

        rc = bcp_init(o_hdbc3, name, NULL, "logs\\ordline.err", DB_IN);
        if (rc != SUCCEEDED)
            HandleErrorDBC(o_hdbc3);

        if ((aptr->build_index == 1) && (aptr->index_order == 1))
        {
            sprintf(bcphint, "tablock, order (ol_w_id, ol_d_id, ol_o_id,
ol_number), ROWS_PER_BATCH = %u", (aptr->num_warehouses * 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);

                hThread[0] = CreateThread(NULL,

0,

(LPTHREAD_START_ROUTINE) LoadOrdersTable,

&orders_time_start,

0,

&dwThreadID[0]);

                if (hThread[0] == NULL)
```

Appendix B - Database Design

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

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

    hThread[1] = CreateThread(NULL,

0,
(LPTHREAD_START_ROUTINE) LoadNewOrderTable,
&new_order_time_start,
0,
&dwThreadID[1]);

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

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

    hThread[2] = CreateThread(NULL,

0,
(LPTHREAD_START_ROUTINE) LoadOrderLineTable,
&order_line_time_start,
0,
&dwThreadID[2]);

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

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

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

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

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

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

    return;
}

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

void OrdersBufInit()
{
    int    i;
    int    j;

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

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

//=====
//
```

Appendix B - Database Design

```
// Function : OrdersBufLoad
//
// Fills shared buffer for ORDERS, NEWORDER, and ORDERLINE
//
//=====
void OrdersBufLoad(int d_id, int w_id)
{
    int    cust[ORDERS_PER_DIST+1];
    long   o_id;
    short  ol;

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

    GetPermutation(cust, ORDERS_PER_DIST);

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

        orders_buf[o_id].o_d_id = d_id;
        orders_buf[o_id].o_w_id = w_id;
        orders_buf[o_id].o_id = o_id+1;
        orders_buf[o_id].o_c_id = cust[o_id+1];
        orders_buf[o_id].o_ol_cnt = (short)RandomNumber(5L, 15L);

        if (o_id < first_new_order)
        {
            orders_buf[o_id].o_carrier_id = (short)RandomNumber(1L,
10L);
            orders_buf[o_id].o_all_local = 1;
        }
        else
        {
            orders_buf[o_id].o_carrier_id = 0;
            orders_buf[o_id].o_all_local = 1;
        }

        for (ol=0; ol<orders_buf[o_id].o_ol_cnt; ol++)
        {
            orders_buf[o_id].o_ol[ol].ol = ol+1;
            orders_buf[o_id].o_ol[ol].ol_i_id = RandomNumber(1L,
max_items);

            orders_buf[o_id].o_ol[ol].ol_supply_w_id = w_id;
            orders_buf[o_id].o_ol[ol].ol_quantity = 5;
            MakeAlphaString(24, 24, OL_DIST_INFO_LEN,
&orders_buf[o_id].o_ol[ol].ol_dist_info);

            // Generate ORDER-LINE data
            if (o_id < first_new_order)
            {
                orders_buf[o_id].o_ol[ol].ol_amount = 0;
                // Added to insure ol_delivery_d set properly

                during load

                FormatDate(&orders_buf[o_id].o_ol[ol].ol_delivery_d);
            }
        }
    }
}
```

```
        else
        {
            orders_buf[o_id].o_ol[ol].ol_amount =
RandomNumber(1,999999)/100.0;
            // Added to insure ol_delivery_d set properly

            during load

            // odbc datetime format

            strcpy(orders_buf[o_id].o_ol[ol].ol_delivery_d,"1899-12-31 12:00:00.000");
        }
    }
}

//=====
//
// Function : LoadOrdersTable
//
//=====
void LoadOrdersTable(LOADER_TIME_STRUCT *orders_time_start)
{
    int    i;
    long   o_id;
    short  o_d_id;
    short  o_w_id;
    long   o_c_id;
    short  o_carrier_id;
    short  o_ol_cnt;
    short  o_all_local;
    char   o_entry_d[O_ENTRY_D_LEN+1];
    RETCODE rc;
    DBINT   rcint;

    // bind ORDER data
    rc = bcp_bind(o_hdbc1, (BYTE *) &o_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

    rc = bcp_bind(o_hdbc1, (BYTE *) &o_c_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc1);

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

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

Appendix B - Database Design

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

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

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

    FormatDate(&o_entry_d);

    // send data to server
    rc = bcp_sendrow(o_hdbc1);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    orders_rows_loaded++;
    CheckForCommit(o_hdbc1, o_hstmt1, orders_rows_loaded, "orders",
&orders_time_start->time_start);
}

// rcint = bcp_batch(o_hdbc1);
// if (rcint < 0)
//     HandleErrorDBC(o_hdbc1);

if ((o_w_id == aptr->num_warehouses) && (o_d_id == 10))
{
    rcint = bcp_done(o_hdbc1);
    if (rcint < 0)
        HandleErrorDBC(o_hdbc1);

    SQLFreeStmt(o_hstmt1, SQL_DROP);
    SQLDisconnect(o_hdbc1);
    SQLFreeConnect(o_hdbc1);

    // if build index after load...
    if ((aptr->build_index == 1) && (aptr->index_order == 0))
        BuildIndex("idxordc1");

    // build non-clustered index
    if (aptr->build_index == 1)
        BuildIndex("idxordnc");
}

}

//=====
//
// Function : LoadNewOrderTable
//
//=====
```

```
void LoadNewOrderTable(LOADER_TIME_STRUCT *new_order_time_start)
{
    int      i;
    long     o_id;
    short    o_d_id;
    short    o_w_id;

    RETCODE  rc;
    DBINT    rcint;

    // Bind NEW-ORDER data

    rc = bcp_bind(o_hdbc2, (BYTE *) &o_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc2);

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

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

    for (i = first_new_order; i < last_new_order; i++)
    {
        o_id  = orders_buf[i].o_id;
        o_d_id = orders_buf[i].o_d_id;
        o_w_id = orders_buf[i].o_w_id;

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

        new_order_rows_loaded++;
        CheckForCommit(o_hdbc2, o_hstmt2, new_order_rows_loaded, "new_order",
&new_order_time_start->time_start);
    }

    // rcint = bcp_batch(o_hdbc2);
    // if (rcint < 0)
    //     HandleErrorDBC(o_hdbc2);

    if ((o_w_id == aptr->num_warehouses) && (o_d_id == 10))
    {
        rcint = bcp_done(o_hdbc2);
        if (rcint < 0)
            HandleErrorDBC(o_hdbc2);

        SQLFreeStmt(o_hstmt2, SQL_DROP);
        SQLDisconnect(o_hdbc2);
        SQLFreeConnect(o_hdbc2);

        // if build index after load...
        if ((aptr->build_index == 1) && (aptr->index_order == 0))
            BuildIndex("idxnodc1");
    }

}

//=====
```

Appendix B - Database Design

```
//
// Function : LoadOrderLineTable
//
//=====
void LoadOrderLineTable(LOADER_TIME_STRUCT *order_line_time_start)
{
    int i,j;
    long o_id;
    short o_d_id;
    short o_w_id;
    long ol;
    long ol_i_id;
    short ol_supply_w_id;
    short ol_quantity;
    double ol_amount;
    char ol_dist_info[DIST_INFO_LEN+1];
    char ol_delivery_d[OL_DELIVERY_D_LEN+1];
    RETCODE rc;
    DBINT rcint;

    // bind ORDER-LINE data
    rc = bcp_bind(o_hdbc3, (BYTE *) &o_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 1);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &o_d_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 2);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &o_w_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2, 3);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &ol, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 4);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &ol_i_id, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT4, 5);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &ol_supply_w_id, 0, SQL_VARLEN_DATA, NULL, 0,
SQLINT2, 6);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &ol_delivery_d, 0, OL_DELIVERY_D_LEN, NULL, 0,
SQLCHARACTER, 7);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &ol_quantity, 0, SQL_VARLEN_DATA, NULL, 0, SQLINT2,
8);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) &ol_amount, 0, SQL_VARLEN_DATA, NULL, 0, SQLFLT8, 9);
    if (rc != SUCCEEDED)
        HandleErrorDBC(o_hdbc3);

    rc = bcp_bind(o_hdbc3, (BYTE *) ol_dist_info, 0, DIST_INFO_LEN, NULL, 0, 0, 10);
    if (rc != SUCCEEDED)
```

```
        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 != SUCCEEDED)
                HandleErrorDBC(o_hdbc3);

            order_line_rows_loaded++;
            CheckForCommit(o_hdbc3, o_hstmt3, order_line_rows_loaded,
"order_line", &order_line_time_start->time_start);
        }

        // rcint = bcp_batch(o_hdbc3);
        // if (rcint < 0)
        //     HandleErrorDBC(o_hdbc3);

        if ((o_w_id == aptr->num_warehouses) && (o_d_id == 10))
        {
            rcint = bcp_done(o_hdbc3);
            if (rcint < 0)
                HandleErrorDBC(o_hdbc3);

            SQLFreeStmt(o_hstmt3, SQL_DROP);
            SQLDisconnect(o_hdbc3);
            SQLFreeConnect(o_hdbc3);

            // if build index after load...
            if ((aptr->build_index == 1) && (aptr->index_order == 0))
                BuildIndex("idxodlcl1");

        }

    }

//=====
//
// Function : GetPermutation
//
//=====

void GetPermutation(int perm[], int n)
{
    int i, r, t;
```

Appendix B - Database Design

```
for (i=1;i<=n;i++)
    perm[i] = i;

for (i=1;i<=n;i++)
{
    r = RandomNumber(i,n);
    t = perm[i];
    perm[i] = perm[r];
    perm[r] = t;
}

//=====
//
// Function   : CheckForCommit
//
//=====

void CheckForCommit(HDBC hdbc,
                   HSTMT hstmt,
                   int rows_loaded,
                   char *table_name,
                   long *time_start)
{
    long time_end, time_diff;
    // DBINT rcint;

    if ( !(rows_loaded % aptr->batch) )
    {
        // rcint = bcp_batch(hdbc);
        // if (rcint < 0)
        //     HandleErrorDBC(hdbc);

        time_end = (TimeNow() / MILLI);
        time_diff = time_end - *time_start;

        printf("> Loaded %ld rows into %s in %ld sec - Total = %d (%.2f
rps)\n",
            aptr->batch,
            table_name,
            time_diff,
            rows_loaded,
            (float) aptr->batch / (time_diff ? time_diff :
1L));

        *time_start = time_end;
    }

    return;
}

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

```
void OpenConnections()
{
    RETCODE rc;

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

    SQLAllocHandle(SQL_HANDLE_ENV, SQL_NULL_HANDLE, &henv );
    SQLSetEnvAttr(henv, SQL_ATTR_ODBC_VERSION, (void*)SQL_OV_ODBC3, 0 );

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

    SQLSetConnectAttr(i_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON, SQL_IS_INTEGER
);
    SQLSetConnectAttr(w_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON, SQL_IS_INTEGER
);
    SQLSetConnectAttr(c_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON, SQL_IS_INTEGER
);
    SQLSetConnectAttr(c_hdbc2, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON, SQL_IS_INTEGER
);
    SQLSetConnectAttr(o_hdbc1, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON, SQL_IS_INTEGER
);
    SQLSetConnectAttr(o_hdbc2, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON, SQL_IS_INTEGER
);
    SQLSetConnectAttr(o_hdbc3, SQL_COPT_SS_BCP, (void *)SQL_BCP_ON, SQL_IS_INTEGER
);

    // Open connections to SQL Server
    // Connection 1

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

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

    rc = SQLDriverConnect ( i_hdbc1,
                            NULL,
                            (SQLCHAR*)&szDriverString[0],
                            SQL_NTS,
                            (SQLCHAR*)&szDriverStringOut[0],
                            sizeof(szDriverStringOut),
                            &cbDriverStringOut,
                            SQL_DRIVER_NOPROMPT );

    if (rc != SUCCEED)
```

Appendix B - Database Design

```
        HandleErrorDBC(i_hdbc1);
// Connection 2
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );

    rc = SQLSetConnectOption (w_hdbc1, SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);

    rc = SQLDriverConnect ( w_hdbc1,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
    );
    if (rc != SUCCEED)
        HandleErrorDBC(w_hdbc1);
// Connection 3
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );

    rc = SQLSetConnectOption (c_hdbc1, SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc1);

    rc = SQLDriverConnect ( c_hdbc1,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
    );
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc1);
// Connection 4
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );

    rc = SQLSetConnectOption (c_hdbc2, SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);

    rc = SQLDriverConnect ( c_hdbc2,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
    );
    if (rc != SUCCEED)
        HandleErrorDBC(c_hdbc2);
// Connection 5
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );

    rc = SQLSetConnectOption (o_hdbc1, SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);

    rc = SQLDriverConnect ( o_hdbc1,
        NULL,
        (SQLCHAR*)&szDriverString[0] ,
        SQL_NTS,
        (SQLCHAR*)&szDriverStringOut[0],
        sizeof(szDriverStringOut),
        &cbDriverStringOut,
        SQL_DRIVER_NOPROMPT
    );
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc1);
// Connection 6
    sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );

    rc = SQLSetConnectOption (o_hdbc2, SQL_PACKET_SIZE, aptr->pack_size);
    if (rc != SUCCEED)
        HandleErrorDBC(o_hdbc2);
```

Appendix B - Database Design

```
rc = SQLDriverConnect ( o_hdbc2,
                        NULL,
                        (SQLCHAR*)&szDriverString[0] ,
                        SQL_NTS,
                        (SQLCHAR*)&szDriverStringOut[0],
                        sizeof(szDriverStringOut),
                        &cbDriverStringOut,
                        SQL_DRIVER_NOPROMPT
);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc2);
// Connection 7
sprintf( szDriverString , "DRIVER={SQL
Server};SERVER=%s;UID=%s;PWD=%s;DATABASE=%s" ,
        aptr->server,
        aptr->user,
        aptr->password,
        aptr->database );
rc = SQLSetConnectOption (o_hdbc3, SQL_PACKET_SIZE, aptr->pack_size);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);
rc = SQLDriverConnect ( o_hdbc3,
                        NULL,
                        (SQLCHAR*)&szDriverString[0] ,
                        SQL_NTS,
                        (SQLCHAR*)&szDriverStringOut[0],
                        sizeof(szDriverStringOut),
                        &cbDriverStringOut,
                        SQL_DRIVER_NOPROMPT
);
if (rc != SUCCEED)
    HandleErrorDBC(o_hdbc3);
}
//=====
//
// Function name: BuildIndex
//
//=====
void BuildIndex(char        *index_script)
{
    char    cmd[256];

    printf("Starting index creation:  %s\n",index_script);

    sprintf(cmd, "isql -S%s -U%s -P%s -e -i%s\\%s.sql > logs\\%s.log",
            aptr->server,
            aptr->user,
            aptr->password,
            aptr->index_script_path,
            index_script,
            index_script);

    system(cmd);

    printf("Finished index creation:  %s\n",index_script);
}

void HandleErrorDBC (SQLHDBC  hdbc1)
{
    SQLCHAR        SqlState[6], Msg[SQL_MAX_MESSAGE_LENGTH];
    SQLINTEGER     NativeError;
    SQLSMALLINT    i, MsgLen;
    SQLRETURN      rc2;
    char           timebuf[128];
    char           datebuf[128];
    FILE           *fp1;

    i = 1;
    while (( rc2 = SQLGetDiagRec(SQL_HANDLE_DBC , hdbc1, i, SqlState ,
    &NativeError,
                                Msg, sizeof(Msg) , &MsgLen ) != SQL_NO_DATA )
    {
        sprintf( szLastError , "%s" , Msg );
        _strtime(timebuf);
        _strdate(datebuf);

        printf( "[%s : %s] %s\n" , datebuf, timebuf, szLastError);

        fp1 = fopen("logs\\tpccldr.err","w");
        if (fp1 == NULL)
            printf("ERROR:  Unable to open errorlog file.\n");
        else
        {
            fprintf(fp1, "[%s : %s] %s\n" , datebuf, timebuf,
            szLastError);
            fclose(fp1);
        }
        i++;
    }
}

void FormatDate ( char* szTimeCOutput )
{
    struct tm when;
    time_t now;

    time( &now );
    when = *localtime( &now );

    mktime( &when );

    // odbc datetime format
    strftime( szTimeCOutput , 30 , "%Y-%m-%d %H:%M:%S.000", &when );
}
```


Appendix B - Database Design

```
        return;
    }
}
```

getargs.c

```
//      File:          GETARGS.C
//      Microsoft TPC-C Kit Ver. 4.00
//      Copyright Microsoft, 1996, 1997, 1998
//      Purpose: Source file for command line processing

// Includes
#include "tpcc.h"

//=====
//
// Function name: GetArgsLoader
//
//=====

void GetArgsLoader(int argc, char **argv, TPCCLDR_ARGS *pargs)
{
    int     i;
    char  *ptr;

#ifdef DEBUG
    printf("[%ld]DBG: Entering GetArgsLoader()\n", (int) GetCurrentThreadId());
#endif

    /* init args struct with some useful values */
    pargs->server      = SERVER;
    pargs->user        = USER;
    pargs->password    = PASSWORD;
    pargs->database    = DATABASE;
    pargs->batch       = BATCH;
    pargs->num_warehouses = UNDEF;
    pargs->tables_all  = TRUE;
    pargs->table_item  = FALSE;
    pargs->table_warehouse = FALSE;
    pargs->table_customer = FALSE;
```

```
    pargs->table_orders      = FALSE;
    pargs->loader_res_file   = LOADER_RES_FILE;
    pargs->pack_size        = DEF_LD_PACK_SIZE;
    pargs->starting_warehouse = DEF_STARTING_WAREHOUSE;
    pargs->build_index       = BUILD_INDEX;
    pargs->index_order       = INDEX_ORDER;
    pargs->index_script_path  = INDEX_SCRIPT_PATH;
    pargs->scale_down        = SCALE_DOWN;
```

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

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

    ptr = argv[i];

    switch ( ptr[1] )
    {
        case 'h': /* Fall through */
        case 'H':
            GetArgsLoaderUsage();
            break;

        case 'D':
            pargs->database = ptr+2;
            break;

        case 'P':
            pargs->password = ptr+2;
            break;

        case 'S':
            pargs->server = ptr+2;
            break;

        case 'U':
            pargs->user = ptr+2;
            break;

        case 'b':
            pargs->batch = atol(ptr+2);
            break;

        case 'W':
            pargs->num_warehouses = atol(ptr+2);
            break;

        case 's':
            pargs->starting_warehouse = atol(ptr+2);
            break;

        case 't':
            {
                pargs->tables_all = FALSE;
                if ( strcmp(ptr+2,"item") == 0 )
```

Appendix B - Database Design

```
0)
TRUE;

        pargs->table_item = TRUE;
    else if (strcmp(ptr+2,"warehouse") ==
        pargs->table_warehouse =
    else if (strcmp(ptr+2,"customer") == 0)
        pargs->table_customer = TRUE;
    else if (strcmp(ptr+2,"orders") == 0)
        pargs->table_orders = TRUE;
    else
    {
        printf("\nUnrecognized command");
        GetArgsLoaderUsage();
        exit(1);
    }
    break;
}

case 'f':
    pargs->loader_res_file = ptr+2;
    break;

case 'p':
    pargs->pack_size = atol(ptr+2);
    break;

case 'i':
    pargs->build_index = atol(ptr+2);
    break;

case 'o':
    pargs->index_order = atol(ptr+2);
    break;

case 'c':
    pargs->scale_down = atol(ptr+2);
    break;

case 'd':
    pargs->index_script_path = ptr+2;
    break;

default:
    GetArgsLoaderUsage();
    exit(-1);
    break;
}

}

/* check for required args */
if (pargs->num_warehouses == UNDEF )
{
    printf("Number of Warehouses is required\n");
    exit(-2);
}

return;
}

//=====
//
```

```
// Function name: GetArgsLoaderUsage
//
//=====

void GetArgsLoaderUsage()
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering GetArgsLoaderUsage()\n", (int) GetCurrentThreadId());
#endif

    printf("TPCCLDR:\n\n");
    printf("Parameter                                     Default\n");
    printf("-----\n");
    printf("-W Number of Warehouses to Load                Required \n");
    printf("-S Server                                         %s\n", SERVER);
    printf("-U Username                                       %s\n", USER);
    printf("-P Password                                       %s\n", PASSWORD);
    printf("-D Database                                       %s\n", DATABASE);
    printf("-b Batch Size                                     %ld\n", (long)
BATCH);
    printf("-p TDS packet size                               %ld\n", (long)
DEFLDPACKSIZE);
    printf("-f Loader Results Output Filename              %s\n",
LOADER_RES_FILE);
    printf("-s Starting Warehouse                           %ld\n", (long)
DEF_STARTING_WAREHOUSE);
    printf("-i Build Option (data = 0, data and index = 1)  %ld\n", (long)
BUILD_INDEX);
    printf("-o Cluster Index Build Order (before = 1, after = 0) %ld\n", (long)
INDEX_ORDER);
    printf("-c Build Scaled Database (normal = 0, tiny = 1) %ld\n", (long)
SCALE_DOWN);
    printf("-d Index Script Path                            %s\n",
INDEX_SCRIPT_PATH);
    printf("-t Table to Load                                 all tables \n");
    printf("    [item|warehouse|customer|orders]\n");
    printf("    Notes: \n");
    printf("    - the '-t' parameter may be included multiple times to \n");
    printf("    - specify multiple tables to be loaded \n");
    printf("    - 'item' loads ITEM table \n");
    printf("    - 'warehouse' loads WAREHOUSE, DISTRICT, and STOCK tables \n");
    printf("    - 'customer' loads CUSTOMER and HISTORY tables \n");
    printf("    - 'orders' load NEW-ORDER, ORDERS, ORDER-LINE tables \n");

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

    exit(0);
}

}

random.c

// File: RANDOM.C
// Microsoft TPC-C Kit Ver. 4.00
// Copyright Microsoft, 1996, 1997, 1998
```

Appendix B - Database Design

```
// Purpose: Random number generation routines for database loader

// Includes
#include "tpcc.h"
#include "math.h"

// Defines
#define A 16807
#define M 2147483647
#define Q 127773 /* M div A */
#define R 2836 /* M mod A */
#define Thread __declspec(thread)

// Globals
long Thread Seed = 0; /* thread local seed */

/*****
 *
 * random -
 * Implements a GOOD pseudo random number generator. This generator
 * will/should? run the complete period before repeating.
 *
 * Copied from:
 * Random Numbers Generators: Good Ones Are Hard to Find.
 * Communications of the ACM - October 1988 Volume 31 Number 10
 *
 * Machine Dependencies:
 * long must be 2 ^ 31 - 1 or greater.
 *****/

/*****
 * seed - load the Seed value used in irand and drand. Should be used before
 * first call to irand or drand.
 *****/

void seed(long val)
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering seed()...\n", (int) GetCurrentThreadId());
    printf("Old Seed %ld New Seed %ld\n",Seed, val);
#endif

    if ( val < 0 )
        val = abs(val);

    Seed = val;
}

/*****
 *
 * irand - returns a 32 bit integer pseudo random number with a period of
 * 1 to 2 ^ 32 - 1.
 *
 * parameters:
 * none.
 *
 * returns:
 * 32 bit integer - defined as long ( see above ).
 *****/
```

```
*
 * side effects:
 * seed get recomputed.
 *****/

long irand()
{
    register long s; /* copy of seed */
    register long test; /* test flag */
    register long hi; /* tmp value for speed */
    register long lo; /* tmp value for speed */

#ifdef DEBUG
    printf("[%ld]DBG: Entering irand()...\n", (int) GetCurrentThreadId());
#endif

    s = Seed;
    hi = s / Q;
    lo = s % Q;

    test = A * lo - R * hi;
    if ( test > 0 )
        Seed = test;
    else
        Seed = test + M;

    return( Seed );
}

/*****
 *
 * drand - returns a double pseudo random number between 0.0 and 1.0.
 * See irand.
 *****/

double drand()
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering drand()...\n", (int) GetCurrentThreadId());
#endif

    return( (double)irand() / 2147483647.0 );
}

//=====
// Function : RandomNumber
//
// Description:
//=====
long RandomNumber(long lower, long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering RandomNumber()...\n", (int) GetCurrentThreadId());
#endif

    if ( upper == lower ) /* pgd 08-13-96 perf enhancement */
        return lower;

    upper++;
```

Appendix B - Database Design

```
    if ( upper <= lower )
        rand_num = upper;
    else
        rand_num = lower + irand() % (upper - lower); /* pgd 08-13-96 perf
enhancement */

#ifdef DEBUG
    printf("[%ld]DBG: RandomNumber between %ld & %ld ==> %ld\n",
           (int) GetCurrentThreadId(), lower, upper,
           rand_num);
#endif

    return rand_num;
}

#if 0
//Original code pgd 08/13/96

long RandomNumber(long lower,
                  long upper)
{
    long rand_num;

#ifdef DEBUG
    printf("[%ld]DBG: Entering RandomNumber()...\n", (int) GetCurrentThreadId());
#endif

    upper++;

    if ((upper <= lower))
        rand_num = upper;
    else
        rand_num = lower + irand() % ((upper > lower) ? upper - lower :
upper);

#ifdef DEBUG
    printf("[%ld]DBG: RandomNumber between %ld & %ld ==> %ld\n",
           (int) GetCurrentThreadId(), lower, upper,
           rand_num);
#endif

    return rand_num;
}
#endif

//=====
// Function : NURand
//
// Description:
//=====
long NURand(int iConst,
            long x,
            long y,
            long C)
{
    long rand_num;
```

```
#ifdef DEBUG
    printf("[%ld]DBG: Entering NURand()...\n", (int) GetCurrentThreadId());
#endif

    rand_num = (((RandomNumber(0,iConst) | RandomNumber(x,y)) + C) % (y-x+1))+x;

#ifdef DEBUG
    printf("[%ld]DBG: NURand: num = %d\n", (int) GetCurrentThreadId(), rand_num);
#endif

    return rand_num;
}
```

strings.c

```
// File: STRINGS.C
// Microsoft TPC-C Kit Ver. 4.00
// Copyright Microsoft, 1996, 1997, 1998
// Purpose: Source file for database loader string functions

// Includes
#include "tpcc.h"
#include <string.h>
#include <ctype.h>

//=====
//
// Function name: MakeAddress
//
//=====

void MakeAddress(char *street_1,
                 char *street_2,
                 char *city,
                 char *state,
                 char *zip)
{
#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeAddress()\n", (int) GetCurrentThreadId());
#endif

    MakeAlphaString (10, 20, ADDRESS_LEN, street_1);
    MakeAlphaString (10, 20, ADDRESS_LEN, street_2);
    MakeAlphaString (10, 20, ADDRESS_LEN, city);
    MakeAlphaString ( 2,  2, STATE_LEN, state);
    MakeZipNumberString( 9,  9, ZIP_LEN, zip);

#ifdef DEBUG
    printf("[%ld]DBG: MakeAddress: street_1: %s, street_2: %s, city: %s, state: %s, zip:
%s\n",
           (int) GetCurrentThreadId(), street_1, street_2, city,
           state, zip);
#endif
}
```

Appendix B - Database Design

```
return;
}

//=====
//
// Function name: LastName
//
//=====

void LastName(int num,
              char *name)
{
    static char *n[] =
    {
        "BAR" , "OUGHT" , "ABLE" , "PRI" , "PRES",
        "ESE" , "ANTI" , "CALLY" , "ATION" , "EING"
    };

#ifdef DEBUG
    printf("[%ld]DBG: Entering LastName()\n", (int) GetCurrentThreadId());
#endif

    if ((num >= 0) && (num < 1000))
    {
        strcpy(name, n[(num/100)%10]);
        strcat(name, n[(num/10)%10]);
        strcat(name, n[(num/1)%10]);

        if (strlen(name) < LAST_NAME_LEN)
        {
            PaddString(LAST_NAME_LEN, name);
        }
    }
    else
    {
        printf("\nError in LastName()... num < %ld> out of range (0,999)\n",
            num);
        exit(-1);
    }

#ifdef DEBUG
    printf("[%ld]DBG: LastName: num = [%d] ==> [%d][%d][%d]\n",
        (int) GetCurrentThreadId(), num, num/100, (num/10)%10,
        num%10);
    printf("[%ld]DBG: LastName: String = %s\n", (int) GetCurrentThreadId(), name);
#endif

    return;
}

//=====
//
// Function name: MakeAlphaString
//
//=====
```

```
//philipdu 08/13/96 Changed MakeAlphaString to use A-Z, a-z, and 0-9 in
//accordance with spec see below:
//The spec says:
//4.3.2.2 The notation random a-string [x .. y]
//(respectively, n-string [x .. y]) represents a string of random alphanumeric
//(respectively, numeric) characters of a random length of minimum x, maximum y,
//and mean (y+x)/2. Alphanumerics are A..Z, a..z, and 0..9. The only other
//requirement is that the character set used "must be able to represent a minimum
//of 128 different characters". We are using 8-bit chars, so this is a non issue.
//It is completely unreasonable to stuff non-printing chars into the text fields.
//-CLevine 08/13/96

int MakeAlphaString( int x, int y, int z, char *str)
{
    int len;
    int i;
    static char chArray[] =
    "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
    static int chArrayMax = 61;

#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeAlphaString()\n", (int) GetCurrentThreadId());
#endif

    len= RandomNumber(x, y);

    for (i=0; i<len; i++)
        str[i] = chArray[RandomNumber(0, chArrayMax)];
    if ( len < z )
        memset(str+len, ' ', z - len);
    str[len] = 0;

    return len;
}

//=====
//
// Function name: MakeOriginalAlphaString
//
//=====

int MakeOriginalAlphaString(int x,
                            int y,
                            int z,
                            char *str,
                            int percent)
{
    int len;
    int val;
    int start;

#ifdef DEBUG
    printf("[%ld]DBG: Entering MakeOriginalAlphaString()\n", (int) GetCurrentThreadId());
#endif

    // verify percentage is valid
    if ((percent < 0) || (percent > 100))
    {
        printf("MakeOriginalAlphaString: Invalid percentage: %d\n", percent);
        exit(-1);
    }
}
```

Appendix B - Database Design

```
// verify string is at least 8 chars in length
if ((x + y) <= 8)
{
    printf("MakeOriginalAlphaString: string length must be >= 8\n");
    exit(-1);
}

// Make Alpha String
len = MakeAlphaString(x,y, z, str);

val = RandomNumber(1,100);
if (val <= percent)
{
    start = RandomNumber(0, len - 8);
    strncpy(str + start, "ORIGINAL", 8);
}

#ifdef DEBUG
printf("[%ld]DBG: MakeOriginalAlphaString: : %s\n",
      (int) GetCurrentThreadId(), str);
#endif

return strlen(str);
}

//=====
//
// Function name: MakeNumberString
//
//=====
int MakeNumberString(int x, int y, int z, char *str)
{
    char tmp[16];

    //MakeNumberString is always called MakeZipNumberString(16, 16, 16, string)

    memset(str, '0', 16);
    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

    itoa(RandomNumber(0, 99999999), tmp, 10);
    memcpy(str+8, tmp, strlen(tmp));

    str[16] = 0;

    return 16;
}

//=====
//
// Function name: MakeZipNumberString
//
//=====
int MakeZipNumberString(int x, int y, int z, char *str)
{
    char tmp[16];

    //MakeZipNumberString is always called MakeZipNumberString(9, 9, 9, string)

    strcpy(str, "000011111");
```

```
    itoa(RandomNumber(0, 9999), tmp, 10);
    memcpy(str, tmp, strlen(tmp));

    return 9;
}

//=====
//
// Function name: InitString
//
//=====
void InitString(char *str, int len)
{
#ifdef DEBUG
printf("[%ld]DBG: Entering InitString()\n", (int) GetCurrentThreadId());
#endif

    memset(str, ' ', len);
    str[len] = 0;
}

//=====
// Function name: InitAddress
//
// Description:
//
//=====
void InitAddress(char *street_1, char *street_2, char *city, char *state, char *zip)
{
    memset(street_1, ' ', ADDRESS_LEN+1);
    memset(street_2, ' ', ADDRESS_LEN+1);
    memset(city, ' ', ADDRESS_LEN+1);

    street_1[ADDRESS_LEN+1] = 0;
    street_2[ADDRESS_LEN+1] = 0;
    city[ADDRESS_LEN+1] = 0;

    memset(state, ' ', STATE_LEN+1);
    state[STATE_LEN+1] = 0;

    memset(zip, ' ', ZIP_LEN+1);
    zip[ZIP_LEN+1] = 0;
}

//=====
//
// Function name: PaddString
//
//=====
void PaddString(int max, char *name)
{
    int len;

    len = strlen(name);
    if ( len < max )
        memset(name+len, ' ', max - len);
    name[max] = 0;
```

Appendix B - Database Design

```
        return;  
    }  
}
```

time.c

```
//      File:          TIME.C  
//      Microsoft TPC-C Kit Ver. 4.00  
//      Copyright Microsoft, 1996, 1997, 1998  
//      Purpose:  Source file for time functions  
  
// Includes  
#include "tpcc.h"  
  
// Globals  
static long start_sec;  
  
//=====  
//  
// Function name: TimeNow  
//  
//=====  
  
long TimeNow()  
{  
    long      time_now;  
    struct _timeb el_time;  
  
#ifdef DEBUG  
    printf("[%ld]DBG: Entering TimeNow()\n", (int) GetCurrentThreadId());  
#endif  
  
    _ftime(&el_time);  
  
    time_now = ((el_time.time - start_sec) * 1000) + el_time.millitm;  
  
    return time_now;  
}
```

Appendix C – Tunable Parameters

Appendix C - Tunable Parameters

Server Configuration Parameters

Microsoft Windows 2003 Server Parameters

The following registry key was added to disable the kernel counters for Global and Per-Process I/Os:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\I/O System]
"CountOperations"=dword:00000000
```

Microsoft Windows 2003 Server Configuration

The following services were disabled on the server:

- Alerter
- Automatic Updates
- Computer Browser
- Cryptographic Services
- DHCP Client
- Distributed File System
- Distributed Link Tracking Client
- DNS Client
- Global Array Manager Server
- Help and Support
- IPSEC Policy Agent
- License Logging Service
- Messenger
- MSSQLserver
- Microsoft Search
- Print Spooler
- Process Control Service
- Remote Registry Service
- Removable Storage
- Run as Service
- System Event Notification
- SSDP Discovery service
- Task Scheduler
- Wireless configuration

Microsoft SQL Server 2000 Startup Parameters

Microsoft SQL Server was started with the following command line options

```
sqlservr -c -x -T3502 -g100
```

where

-c	Start SQL Server independently of the Microsoft Windows NT Service Control Manager.
----	-------------------------------------------------------------------------------------

Appendix C – Tunable Parameters

-x	Disable the keeping of CPU time and cache-hit ratio statistics.
-T3502	Prints a message to the log at the beginning and end of each checkpoint.
-g150	Reserve 150 MB for non-buffer pool allocations

Microsoft SQL Server Stack Size

The default stack size of Microsoft SQL Server was changed using the EDITBIN utility. The EDITBIN utility ships with Microsoft Visual C++. The command used was editbin /stack:131072 sqlservr.exe.

Microsoft SQL Server 2000 Configuration Parameters

name	minimum	maximum	config_value	run_value
affinity mask	0	2147483647	3	3
allow updates	0	1	1	1
c2 audit mode	0	1	0	0
cost threshold for parallelism	0	32767	5	5
cursor threshold	-1	2147483647	-1	-1
default full-text language	0	2147483647	1033	1033
default language	0	9999	0	0
fill factor (%)	0	100	0	0
index create memory (KB)	704	1600000	0	0
language in cache	3	100	3	3
lightweight pooling	0	1	1	1
locks	5000	2147483647	0	0
max degree of parallelism	0	32	1	1
max server memory (MB)	4	2147483647	2000	2000
max text repl size (B)	0	2147483647	65536	65536
max worker threads	10	1024	320	320
media retention	0	365	0	0
min memory per query (KB)	512	2147483647	1024	1024
min server memory (MB)	0	2147483647	0	0
nested triggers	0	1	1	1
network packet size (B)	512	65535	4096	4096
open objects	0	2147483647	0	0
priority boost	0	1	1	1
query governor cost limit	0	2147483647	0	0
query wait (s)	-1	2147483647	-1	-1
recovery interval (min)	0	32767	32767	32767
remote access	0	1	0	0
remote login timeout (s)	0	2147483647	5	5
remote proc trans	0	1	0	0
remote query timeout (s)	0	2147483647	0	0
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	16383	0	0

Appendix C – Tunable Parameters

System Information report written at: 01/09/04 14:54:17

System Name: PE2650

[System Summary]

Item Value
OS Name Microsoft(R) Windows(R) Server 2003, Standard Edition
Version 5.2.3790 Build 3790
OS Manufacturer Microsoft Corporation
System Name PE2650
System Manufacturer Dell Computer Corporation
System Model PowerEdge 2650
System Type X86-based PC
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~3189 Mhz
Processor x86 Family 15 Model 2 Stepping 5 GenuineIntel ~3189 Mhz
BIOS Version/Date Dell Computer Corporation X46, 6/23/2003
SMBIOS Version 2.3
Windows Directory C:\WINDOWS
System Directory C:\WINDOWS\system32
Boot Device \Device\HarddiskVolume1
Locale United States
Hardware Abstraction Layer Version = "5.2.3790.0 (srv03_rtm.030324-2048)"
User Name PE2650\Administrator
Time Zone Central Standard Time
Total Physical Memory 2,560.00 MB
Available Physical Memory 837.16 MB
Total Virtual Memory 6.85 GB
Available Virtual Memory 3.19 GB
Page File Space 4.35 GB
Page File C:\pagefile.sys

[Hardware Resources]

[Conflicts/Sharing]

Resource	Device
Memory Address 0xF0000000-0xF7FFFFFF	PCI bus
Memory Address 0xF0000000-0xF7FFFFFF	DELL PERC 3/DC Plus RAID Controller
IRQ 30 Dell PERC 3 RAID (SCSI chip)	
IRQ 30 Dell PERC 3/Di RAID Controller	
I/O Port 0x00000000-0x000003AF	PCI bus
I/O Port 0x00000000-0x000003AF	Direct memory access controller
Memory Address 0xFD000000-0xFEBFFFFFF	PCI bus
Memory Address 0xFD000000-0xFEBFFFFFF	RAGE XL PCI Family (Microsoft Corporation)
Memory Address 0xE0000000-0xEFDFEFFF	PCI bus
Memory Address 0xE0000000-0xEFDFEFFF	Dell PERC 3/Di RAID Controller
Memory Address 0xA0000-0xBFFFF	PCI bus
Memory Address 0xA0000-0xBFFFF	RAGE XL PCI Family (Microsoft Corporation)

Appendix C – Tunable Parameters

Memory Address 0xF8000000-0xFCFFFFFF PCI bus
Memory Address 0xF8000000-0xFCFFFFFF DELL PERC 3/DC Plus RAID Controller

I/O Port 0x000003B0-0x000003DF PCI bus
I/O Port 0x000003B0-0x000003DF RAGE XL PCI Family (Microsoft Corporation)

I/O Port 0x0000C000-0x0000CFFF PCI bus
I/O Port 0x0000C000-0x0000CFFF PCI standard PCI-to-PCI bridge

[DMA]

Resource	Device	Status
Channel 4	Direct memory access controller	OK
Channel 2	Standard floppy disk controller	OK

[Forced Hardware]

Device PNP Device ID

[I/O]

Resource	Device	Status
0x00000000-0x000003AF	PCI bus	OK
0x00000000-0x000003AF	Direct memory access controller	OK
0x000003B0-0x000003DF	PCI bus	OK
0x000003B0-0x000003DF	RAGE XL PCI Family (Microsoft Corporation)	OK
0x000003E0-0x00000CF7	PCI bus	OK
0x00000D00-0x00000FFF	PCI bus	OK
0x0000E000-0x0000EFFF	PCI bus	OK
0x0000ECF8-0x0000ECFF	PCI Device	OK
0x0000ECE8-0x0000ECEF	PCI Device	OK
0x0000EC80-0x0000ECBF	PCI Device	OK
0x0000ECF4-0x0000ECF7	PCI Device	OK
0x0000E800-0x0000E8FF	RAGE XL PCI Family (Microsoft Corporation)	OK
0x000003C0-0x000003DF	RAGE XL PCI Family (Microsoft Corporation)	OK
0x00000080-0x0000009F	Direct memory access controller	OK
0x000000C0-0x000000DF	Direct memory access controller	OK
0x0000040B-0x0000040B	Direct memory access controller	OK
0x000004D6-0x000004D6	Direct memory access controller	OK
0x000000F0-0x000000FF	Numeric data processor	OK
0x00000020-0x0000003F	Programmable interrupt controller	OK
0x000000A0-0x000000BF	Programmable interrupt controller	OK
0x000004D0-0x000004D1	Programmable interrupt controller	OK
0x00000061-0x00000061	System speaker	OK
0x00000040-0x0000005F	System timer	OK
0x000003F0-0x000003F5	Standard floppy disk controller	OK
0x000003F7-0x000003F7	Standard floppy disk controller	OK
0x00000060-0x00000060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x00000064-0x00000064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x000003F8-0x000003FF	Communications Port (COM2)	OK
0x000002F8-0x000002FF	Communications Port (COM1)	OK
0x00000070-0x0000007F	System CMOS/real time clock	OK

Appendix C – Tunable Parameters

0x00000800-0x0000089F	System board	OK
0x000008A0-0x000008AF	System board	OK
0x00000C00-0x00000CD7	System board	OK
0x00000F50-0x00000F58	System board	OK
0x000008E0-0x000008E3	System board	OK
0x000000E0-0x000000EF	System board	OK
0x000008B0-0x000008BF	CSB5 IDE Controller	OK
0x000001F0-0x000001F7	Primary IDE Channel	OK
0x000003F6-0x000003F6	Primary IDE Channel	OK
0x00000170-0x00000177	Secondary IDE Channel	OK
0x00000376-0x00000376	Secondary IDE Channel	OK
0x00000A79-0x00000A79	ISAPNP Read Data Port	OK
0x00000279-0x00000279	ISAPNP Read Data Port	OK
0x00000274-0x00000277	ISAPNP Read Data Port	OK
0x0000C000-0x0000CFFF	PCI bus	OK
0x0000C000-0x0000CFFF	PCI standard PCI-to-PCI bridge	OK
0x0000CC00-0x0000CCFF	Dell PERC 3 RAID (SCSI chip)	OK
0x0000C800-0x0000C8FF	Dell PERC 3 RAID (SCSI chip)	OK
0x0000D000-0x0000DFFF	PCI bus	OK

[IRQs]

Resource	Device	Status
IRQ 9	Microsoft ACPI-Compliant System	OK
IRQ 11	PCI Device	OK
IRQ 10	PCI Device	OK
IRQ 7	PCI Device	OK
IRQ 13	Numeric data processor	OK
IRQ 0	System timer	OK
IRQ 6	Standard floppy disk controller	OK
IRQ 1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
IRQ 12	PS/2 Compatible Mouse	OK
IRQ 4	Communications Port (COM2)	OK
IRQ 3	Communications Port (COM1)	OK
IRQ 8	System CMOS/real time clock	OK
IRQ 14	Primary IDE Channel	OK
IRQ 5	ServerWorks (RCC) PCI to USB Open Host Controller	OK
IRQ 30	Dell PERC 3 RAID (SCSI chip)	OK
IRQ 30	Dell PERC 3/Di RAID Controller	OK
IRQ 31	Dell PERC 3 RAID (SCSI chip)	OK
IRQ 28	Broadcom NetXtreme Gigabit Ethernet #3	OK
IRQ 29	Broadcom NetXtreme Gigabit Ethernet #4	OK
IRQ 24	DELL PERC 3/DC Plus RAID Controller	OK
IRQ 20	DELL PERC 3/DC Plus RAID Controller	OK

[Memory]

Resource	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	RAGE XL PCI Family (Microsoft Corporation)	OK
0xD0000-0xE7FFF	PCI bus	OK
0xFD000000-0xFEBFFFF	PCI bus	OK
0xFD000000-0xFEBFFFF	RAGE XL PCI Family (Microsoft Corporation)	OK
0xFEB80000-0xFEB80FFF	PCI Device	OK
0xFE102000-0xFE102FFF	PCI Device	OK

Appendix C – Tunable Parameters

0xFEB00000-0xFEB7FFFF	PCI Device	OK
0xFE101000-0xFE101FFF	RAGE XL PCI Family (Microsoft Corporation)	OK
0xFE100000-0xFE100FFF	ServerWorks (RCC) PCI to USB Open Host Controller	OK
0xE0000000-0xEFDFEFFF	PCI bus	OK
0xE0000000-0xEFDFEFFF	Dell PERC 3/Di RAID Controller	OK
0xEFC00000-0xEFDFEFFF	PCI standard PCI-to-PCI bridge	OK
0xEFCFF000-0xEFCFFFFF	Dell PERC 3 RAID (SCSI chip)	OK
0xEFCFE000-0xEFCFEFFF	Dell PERC 3 RAID (SCSI chip)	OK
0xEFE00000-0xEFFFFFFF	PCI bus	OK
0xEFF10000-0xEFF1FFFF	Broadcom NetXtreme Gigabit Ethernet #3	OK
0xEFF00000-0xEFF0FFFF	Broadcom NetXtreme Gigabit Ethernet #4	OK
0xF0000000-0xF7FFFFFF	PCI bus	OK
0xF0000000-0xF7FFFFFF	DELL PERC 3/DC Plus RAID Controller	OK
0xF8000000-0xFCFFFFFF	PCI bus	OK
0xF8000000-0xFCFFFFFF	DELL PERC 3/DC Plus RAID Controller	OK

[Components]

[Multimedia]

[Audio Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size	Creation Date
c:\windows\system32\msg711.acm	Microsoft Corporation		OK				
		C:\WINDOWS\system32\MSG711.ACM		5.2.3790.0 (srv03_rtm.030324-2048)		10.00 KB (10,240 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msadp32.acm	Microsoft Corporation		OK				
		C:\WINDOWS\system32\MSADP32.ACM		5.2.3790.0 (srv03_rtm.030324-2048)		14.50 KB (14,848 bytes)	3/29/2003 12:00 AM
c:\windows\system32\tsssoft32.acm	DSP GROUP, INC.		OK				
		C:\WINDOWS\system32\TSSOFT32.ACM		1.01		9.50 KB (9,728 bytes)	3/29/2003 12:00 AM
c:\windows\system32\sl_anet.acm	Sipro Lab Telecom Inc.	Sipro Lab Telecom Audio Codec	OK				
		C:\WINDOWS\system32\SL_ANET.ACM		3.02		84.00 KB (86,016 bytes)	3/29/2003 12:00 AM
c:\windows\system32\l3codeca.acm	Fraunhofer Institut Integrierte Schaltungen IIS	Fraunhofer IIS MPEG Layer-3 Codec	OK				
		C:\WINDOWS\system32\L3CODECA.ACM		1, 9, 0, 0305		284.00 KB (290,816 bytes)	3/29/2003 12:00 AM
c:\windows\system32\imaadp32.acm	Microsoft Corporation		OK				
		C:\WINDOWS\system32\IMAADP32.ACM		5.2.3790.0 (srv03_rtm.030324-2048)		15.50 KB (15,872 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msgsm32.acm	Microsoft Corporation		OK				
		C:\WINDOWS\system32\MSGSM32.ACM		5.2.3790.0 (srv03_rtm.030324-2048)		20.50 KB (20,992 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msaud32.acm	Microsoft Corporation	Windows Media Audio Codec	OK				
		C:\WINDOWS\system32\MSAUD32.ACM		8.00.00.4487		288.00 KB (294,912 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msg723.acm	Microsoft Corporation		OK				
		C:\WINDOWS\system32\MSG723.ACM		4.4.4000		116.00 KB (118,784 bytes)	5/30/2003 3:10 PM

Appendix C – Tunable Parameters

[Video Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size	Creation Date
c:\windows\system32\msh261.drv	Microsoft Corporation					OK	
	C:\WINDOWS\system32\MSH261.DRV	4.4.4000				180.00 KB (184,320 bytes)	5/30/2003 3:10 PM
c:\windows\system32\tsbyuv.dll	Microsoft Corporation					OK	
	C:\WINDOWS\system32\TSBYUV.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				8.00 KB (8,192 bytes)	3/24/2003 7:50 PM
c:\windows\system32\msrle32.dll	Microsoft Corporation					OK	
	C:\WINDOWS\system32\MSRLE32.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				10.50 KB (10,752 bytes)	3/29/2003 12:00 AM
c:\windows\system32\iyuv_32.dll	Microsoft Corporation					OK	
	C:\WINDOWS\system32\IYUV_32.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				45.00 KB (46,080 bytes)	3/24/2003 7:49 PM
c:\windows\system32\msvidc32.dll	Microsoft Corporation					OK	
	C:\WINDOWS\system32\MSVIDC32.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				26.50 KB (27,136 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msyuv.dll	Microsoft Corporation					OK	
	C:\WINDOWS\system32\MSYUV.DLL	5.2.3790.0 (srv03_rtm.030324-2048)				16.50 KB (16,896 bytes)	3/24/2003 7:49 PM
c:\windows\system32\msh263.drv	Microsoft Corporation					OK	
	C:\WINDOWS\system32\MSH263.DRV	4.4.4000				284.00 KB (290,816 bytes)	3/24/2003 7:46 PM

[CD-ROM]

Item	Value
Drive	D:
Description	CD-ROM Drive
Media Loaded	No
Media Type	CD-ROM
Name	SAMSUNG CD-ROM SN-124
Manufacturer	(Standard CD-ROM drives)
Status	OK
Transfer Rate	Not Available
SCSI Target ID	0
PNP Device ID	IDE\CDROMSAMSUNG_CD-ROM_SN-124_____Q009____\5&3125DC91&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), 3/29/2003 12:00 AM)

[Sound Device]

Item	Value
------	-------

[Display]

Item	Value
Name	RAGE XL PCI Family (Microsoft Corporation)
PNP Device ID	PCI\VEN_1002&DEV_4752&SUBSYS_01211028&REV_27\3&13C0B0C5&0&70
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)

Appendix C – Tunable Parameters

Installed Driversati2drad.dll
Driver Version 5.10.3663.6013
INF Fileatiixpad.inf (ati2mpad section)
Color Planes 1
Color Table Entries 65536
Resolution 1024 x 768 x 60 hertz
Bits/Pixel 16
Memory Address 0xFD000000-0xFEBFFFFF
I/O Port 0x0000E800-0x0000E8FF
Memory Address 0xFE101000-0xFE101FFF
I/O Port 0x000003B0-0x000003DF
I/O Port 0x000003C0-0x000003DF
Memory Address 0xA0000-0xBFFFFF
Driver c:\windows\system32\drivers\ati2mpad.sys (5.10.3663.6013, 335.38 KB (343,424 bytes), 5/30/2003 10:02 AM)

[Infrared]

Item Value

[Input]

[Keyboard]

Item Value
Description Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Name Enhanced (101- or 102-key)
Layout 00000409
PNP Device ID ACPI\PNP0303\4&25F73A82&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), 3/29/2003 12:00 AM)

[Pointing Device]

Item Value
Hardware Type PS/2 Compatible Mouse
Number of Buttons 3
Status OK
PNP Device ID ACPI\PNP0F13\4&25F73A82&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), 3/29/2003 12:00 AM)

[Modem]

Item Value

Appendix C – Tunable Parameters

[Network]

[Adapter]

Item Value
Name [00000001] Broadcom NetXtreme Gigabit Ethernet
Adapter Type Not Available
Product Type Broadcom NetXtreme Gigabit Ethernet
Installed Yes
PNP Device ID Not Available
Last Reset 1/8/2004 4:54 PM
Index 1
Service Name b57w2k
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] Broadcom NetXtreme Gigabit Ethernet
Adapter Type Not Available
Product Type Broadcom NetXtreme Gigabit Ethernet
Installed Yes
PNP Device ID Not Available
Last Reset 1/8/2004 4:54 PM
Index 2
Service Name b57w2k
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available

Name [00000003] RAS Async Adapter
Adapter Type Not Available
Product Type RAS Async Adapter
Installed Yes
PNP Device ID Not Available
Last Reset 1/8/2004 4:54 PM
Index 3
Service Name AsyncMac
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available

Appendix C – Tunable Parameters

DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address Not Available

Name [00000004] WAN Miniport (L2TP)
Adapter Type Not Available
Product Type WAN Miniport (L2TP)
Installed Yes
PNP Device ID ROOT\MS_L2TPMINIPORT\0000
Last Reset 1/8/2004 4:54 PM
Index 4
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), 3/29/2003 12:00 AM)

Name [00000005] 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 1/8/2004 4:54 PM
Index 5
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\rasppt.sys (5.2.3790.0 (srv03_rtm.030324-2048), 70.50 KB (72,192 bytes), 3/29/2003 12:00 AM)

Name [00000006] 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 1/8/2004 4:54 PM
Index 6
Service Name RasPppoe
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available

Appendix C – Tunable Parameters

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), 3/29/2003 12:00 AM)

Name [00000007] Direct Parallel
Adapter Type Not Available
Product Type Direct Parallel
Installed Yes
PNP Device ID ROOT\MS_PTMINIPORT\0000
Last Reset 1/8/2004 4:54 PM
Index 7
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), 3/29/2003 12:00 AM)

Name [00000008] WAN Miniport (IP)
Adapter Type Not Available
Product Type WAN Miniport (IP)
Installed Yes
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 1/8/2004 4:54 PM
Index 8
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), 3/29/2003 12:00 AM)

Name [00000009] Broadcom NetXtreme Gigabit Ethernet
Adapter Type Ethernet 802.3
Product Type Broadcom NetXtreme Gigabit Ethernet
Installed Yes
PNP Device ID
PCI\VEN_14E4&DEV_16A7&SUBSYS_01211028&REV_02\3&172E68DD&0&30
Last Reset 1/8/2004 4:54 PM
Index 9
Service Name b57w2k
IP Address 192.1.10.78
IP Subnet 255.255.255.0

Appendix C – Tunable Parameters

Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available
DHCP Lease Expires Not Available
DHCP Lease Obtained Not Available
MAC Address 00:06:5B:F8:5D:AA
Memory Address 0xEFF10000-0xEFF1FFFF
IRQ Channel IRQ 28
Driver c:\windows\system32\drivers\b57xp32.sys (2.91.0.0 built by: WinDDK, 137.00 KB (140,288 bytes), 5/30/2003 10:02 AM)

Name [00000010] Broadcom NetXtreme Gigabit Ethernet
Adapter Type Ethernet 802.3
Product Type Broadcom NetXtreme Gigabit Ethernet
Installed Yes
PNP Device ID
PCIIVEN_14E4&DEV_16A7&SUBSYS_01211028&REV_02\3&172E68DD&0&40
Last Reset 1/8/2004 4:54 PM
Index 10
Service Name b57w2k
IP Address 192.1.1.78
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 00:06:5B:F8:5D:AB
Memory Address 0xEFF00000-0xEFF0FFFF
IRQ Channel IRQ 29
Driver c:\windows\system32\drivers\b57xp32.sys (2.91.0.0 built by: WinDDK, 137.00 KB (140,288 bytes), 5/30/2003 10:02 AM)

[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]

Appendix C – Tunable Parameters

Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 16 bytes
Maximum Message Size 63.93 KB (65,467 bytes)
Message Oriented Yes
Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting Yes

Name RSVP UDP Service Provider

Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 16 bytes
Maximum Message Size 63.93 KB (65,467 bytes)
Message Oriented Yes
Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption Yes
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting Yes

Name RSVP TCP Service Provider

Connectionless Service No
Guarantees Delivery Yes
Guarantees SequencingYes
Maximum Address Size 16 bytes
Maximum Message Size 0 bytes
Message Oriented No
Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption Yes
Supports Expedited Data Yes
Supports Graceful Closing Yes
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\\Device\NetBT_Tcpip_{2EF7CB45-6E02-4028-A9B2-7207ECD6BD5B}] SEQPACKET 5

Appendix C – Tunable Parameters

Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{2EF7CB45-6E02-4028-A9B2-7207ECD6BD5B}] DATAGRAM 5

Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{D6D57ED7-08E8-4E49-AE34-29F7F2B4AC24}] SEQPACKET 4

Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Appendix C – Tunable Parameters

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{D6D57ED7-08E8-4E49-AE34-29F7F2B4AC24}] DATAGRAM 4
Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{6068DFB2-35CC-4334-A1CD-99BF402C2EB1}] SEQPACKET 0
Connectionless Service No
Guarantees Delivery Yes
Guarantees SequencingYes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{6068DFB2-35CC-4334-A1CD-99BF402C2EB1}] DATAGRAM 0
Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No

Appendix C – Tunable Parameters

Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{84CBEDAE-78F6-4C0F-A714-0B9A6BAD30DA}] SEQPACKET 1

Connectionless Service No

Guarantees Delivery Yes

Guarantees Sequencing Yes

Maximum Address Size 20 bytes

Maximum Message Size 62.50 KB (64,000 bytes)

Message Oriented Yes

Minimum Address Size 20 bytes

Pseudo Stream Oriented No

Supports Broadcasting No

Supports Connect Data No

Supports Disconnect Data No

Supports Encryption No

Supports Expedited Data No

Supports Graceful Closing No

Supports Guaranteed Bandwidth No

Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{84CBEDAE-78F6-4C0F-A714-0B9A6BAD30DA}] DATAGRAM 1

Connectionless Service Yes

Guarantees Delivery No

Guarantees Sequencing No

Maximum Address Size 20 bytes

Maximum Message Size 62.50 KB (64,000 bytes)

Message Oriented Yes

Minimum Address Size 20 bytes

Pseudo Stream Oriented No

Supports Broadcasting Yes

Supports Connect Data No

Supports Disconnect Data No

Supports Encryption No

Supports Expedited Data No

Supports Graceful Closing No

Supports Guaranteed Bandwidth No

Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{D5B83B02-01EF-4E50-9720-1B8E810DE871}] SEQPACKET 2

Connectionless Service No

Guarantees Delivery Yes

Guarantees Sequencing Yes

Maximum Address Size 20 bytes

Maximum Message Size 62.50 KB (64,000 bytes)

Message Oriented Yes

Minimum Address Size 20 bytes

Pseudo Stream Oriented No

Supports Broadcasting No

Supports Connect Data No

Supports Disconnect Data No

Supports Encryption No

Supports Expedited Data No

Appendix C – Tunable Parameters

Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [Device\NetBT_Tcpip_{D5B83B02-01EF-4E50-9720-1B8E810DE871}]
DATAGRAM 2

Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [Device\NetBT_Tcpip_{AA843776-1681-4379-B2C9-EC445DC298EF}] SEQPACKET 3

Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [Device\NetBT_Tcpip_{AA843776-1681-4379-B2C9-EC445DC298EF}] DATAGRAM 3

Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No

Appendix C – Tunable Parameters

Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

[WinSock]

Item Value
File c:\windows\system32\winsock.dll
Size 2.80 KB (2,864 bytes)
Version 3.10

File c:\windows\system32\wsock32.dll
Size 22.00 KB (22,528 bytes)
Version 5.2.3790.0 (srv03_rtm.030324-2048)

[Ports]

[Serial]

Item Value
Name Communications Port (COM2)
Status OK
PNP Device ID ACPI\PNP0501\1
Maximum Input Buffer Size 0
Maximum Output Buffer Size No
Settable Baud Rate Yes
Settable Data Bits Yes
Settable Flow Control Yes
Settable Parity Yes
Settable Parity Check Yes
Settable Stop Bits Yes
Settable RLSD Yes
Supports RLSD Yes
Supports 16 Bit Mode No
Supports Special Characters No
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy No
Abort Read/Write on Error No
Binary Mode Enabled Yes
Continue XMit on XOff No
CTS Outflow Control No
Discard NULL Bytes No
DSR Outflow Control 0
DSR Sensitivity 0
DTR Flow Control Type Enable
EOF Character 0
Error Replace Character 0
Error Replacement Enabled No

Appendix C – Tunable Parameters

Event Character 0
Parity Check Enabled No
RTS Flow Control Type Enable
XOff Character 19
XOffXMit Threshold 512
XOn Character 17
XOnXMit Threshold 2048
XOnXOff InFlow Control0
XOnXOff OutFlow Control 0
I/O Port 0x000003F8-0x000003FF
IRQ Channel IRQ 4
Driver c:\windows\system32\drivers\serial.sys (5.2.3790.0 (srv03_rtm.030324-2048), 76.00 KB (77,824 bytes), 3/29/2003 12:00 AM)

Name Communications Port (COM1)
Status OK
PNP Device ID ACPI\PNP0501\2
Maximum Input Buffer Size 0
Maximum Output Buffer Size No
Settable Baud Rate Yes
Settable Data Bits Yes
Settable Flow Control Yes
Settable Parity Yes
Settable Parity Check Yes
Settable Stop Bits Yes
Settable RLSD Yes
Supports RLSD Yes
Supports 16 Bit Mode No
Supports Special Characters No
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy No
Abort Read/Write on Error No
Binary Mode Enabled Yes
Continue XMit on XOff No
CTS Outflow Control No
Discard NULL Bytes No
DSR Outflow Control 0
DSR Sensitivity 0
DTR Flow Control Type Enable
EOF Character 0
Error Replace Character 0
Error Replacement Enabled No
Event Character 0
Parity Check Enabled No
RTS Flow Control Type Enable
XOff Character 19
XOffXMit Threshold 512
XOn Character 17
XOnXMit Threshold 2048
XOnXOff InFlow Control0
XOnXOff OutFlow Control 0
I/O Port 0x000002F8-0x000002FF

Appendix C – Tunable Parameters

IRQ Channel IRQ 3

Driver c:\windows\system32\drivers\serial.sys (5.2.3790.0 (srv03_rtm.030324-2048), 76.00 KB (77,824 bytes), 3/29/2003 12:00 AM)

[Parallel]

Item Value

[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 8.79 GB (9,434,361,856 bytes)

Free Space 3.44 GB (3,694,940,160 bytes)

Volume Name

Volume Serial Number 18A97566

Drive D:

Description CD-ROM Disc

Drive K:

Description Local Fixed Disk

Compressed Not Available

File System Not Available

Size Not Available

Free Space Not Available

Volume Name Not Available

Volume Serial Number Not Available

Drive L:

Description Local Fixed Disk

Compressed Not Available

File System Not Available

Size Not Available

Free Space Not Available

Volume Name Not Available

Volume Serial Number Not Available

Drive S:

Description Local Fixed Disk

Compressed Not Available

File System Not Available

Size Not Available

Free Space Not Available

Volume Name Not Available

Appendix C – Tunable Parameters

Volume Serial Number Not Available

Drive U:

Description Local Fixed Disk
Compressed No
File System NTFS
Size 159.93 GB (171,719,131,136 bytes)
Free Space 95.41 GB (102,441,680,896 bytes)
Volume Name Backup1
Volume Serial Number C47CFDC0

Drive V:

Description Local Fixed Disk
Compressed No
File System NTFS
Size 159.93 GB (171,719,131,136 bytes)
Free Space 95.41 GB (102,441,746,432 bytes)
Volume Name Backup2
Volume Serial Number 6C8D077B

Drive W:

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

Drive Y:

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

[Disks]

Item Value
Description Disk drive
Manufacturer (Standard disk drives)
Model DELL Container SCSI Disk Device
Bytes/Sector 512
Media Loaded Yes
Media Type Fixed hard disk
Partitions 2
SCSI Bus 4
SCSI Logical Unit 0
SCSI Port 6
SCSI Target ID 0
Sectors/Track 63
Size 67.79 GB (72,793,728,000 bytes)
Total Cylinders 8,850

Appendix C – Tunable Parameters

Total Sectors 142,175,250
Total Tracks 2,256,750
Tracks/Cylinder 255
Partition Disk #2, Partition #0
Partition Size 8.79 GB (9,434,363,904 bytes)
Partition Starting Offset 32,256 bytes
Partition Disk #2, Partition #1
Partition Size 59.01 GB (63,359,331,840 bytes)
Partition Starting Offset 9,434,396,160 bytes

Description Disk drive
Manufacturer (Standard disk drives)
Model PERC LD 0 PERCRAID SCSI Disk Device
Bytes/Sector 512
Media Loaded Yes
Media Type Fixed hard disk
Partitions 3
SCSI Bus 3
SCSI Logical Unit 0
SCSI Port 3
SCSI Target ID 0
Sectors/Track 63
Size 472.44 GB (507,277,693,440 bytes)
Total Cylinders 61,673
Total Sectors 990,776,745
Total Tracks 15,726,615
Tracks/Cylinder 255
Partition Disk #1, Partition #0
Partition Size 156.26 GB (167,779,229,184 bytes)
Partition Starting Offset 32,256 bytes
Partition Disk #1, Partition #1
Partition Size 156.26 GB (167,779,261,440 bytes)
Partition Starting Offset 167,779,261,440 bytes
Partition Disk #1, Partition #2
Partition Size 159.93 GB (171,719,170,560 bytes)
Partition Starting Offset 335,558,522,880 bytes

Description Disk drive
Manufacturer (Standard disk drives)
Model PERC LD 0 PERCRAID SCSI Disk Device
Bytes/Sector 512
Media Loaded Yes
Media Type Fixed hard disk
Partitions 3
SCSI Bus 3
SCSI Logical Unit 0
SCSI Port 2
SCSI Target ID 0
Sectors/Track 63
Size 472.44 GB (507,277,693,440 bytes)
Total Cylinders 61,673
Total Sectors 990,776,745
Total Tracks 15,726,615
Tracks/Cylinder 255
Partition Disk #0, Partition #0

Appendix C – Tunable Parameters

Partition Size 156.26 GB (167,779,229,184 bytes)
Partition Starting Offset 32,256 bytes
Partition Disk #0, Partition #1
Partition Size 156.26 GB (167,779,261,440 bytes)
Partition Starting Offset 167,779,261,440 bytes
Partition Disk #0, Partition #2
Partition Size 159.93 GB (171,719,170,560 bytes)
Partition Starting Offset 335,558,522,880 bytes

[SCSI]

Item Value

Name Dell PERC 3 RAID (SCSI chip)

Manufacturer Dell

Status OK

PNP Device ID

PCIIVEN_9005&DEV_00C5&SUBSYS_00C51028&REV_01\4&22300438&0&3040

I/O Port 0x0000CC00-0x0000CCFF

Memory Address 0xEFCFF000-0xEFCFFFFF

IRQ Channel IRQ 30

Driver c:\windows\system32\drivers\perc2.sys (5.2.3646.0 (Lab01_N(portbld).020612-1346), 27.63 KB (28,288 bytes), 3/29/2003 12:00 AM)

Name Dell PERC 3 RAID (SCSI chip)

Manufacturer Dell

Status OK

PNP Device ID

PCIIVEN_9005&DEV_00C5&SUBSYS_00C51028&REV_01\4&22300438&0&3140

I/O Port 0x0000C800-0x0000C8FF

Memory Address 0xEFCFE000-0xEFCFEFFF

IRQ Channel IRQ 31

Driver c:\windows\system32\drivers\perc2.sys (5.2.3646.0 (Lab01_N(portbld).020612-1346), 27.63 KB (28,288 bytes), 3/29/2003 12:00 AM)

Name Dell PERC 3/Di RAID Controller

Manufacturer Dell

Status OK

PNP Device ID PCIIVEN_1028&DEV_000A&SUBSYS_01211028&REV_01\3&474B838&0&41

Memory Address 0xE0000000-0xEFDFFFFF

IRQ Channel IRQ 30

Driver c:\windows\system32\drivers\perc2.sys (5.2.3646.0 (Lab01_N(portbld).020612-1346), 27.63 KB (28,288 bytes), 3/29/2003 12:00 AM)

Name DELL PERC 3/DC Plus RAID Controller

Manufacturer DELL

Status OK

PNP Device ID PCIIVEN_101E&DEV_1960&SUBSYS_04941028&REV_01\3&29E81982&0&30

Memory Address 0xF0000000-0xF7FFFFFF

IRQ Channel IRQ 24

Driver c:\windows\system32\drivers\mraid35x.sys (5.2.22.4 built by: WinDDK, 17.88 KB (18,304 bytes), 1/9/2003 11:54 AM)

Name DELL PERC 3/DC Plus RAID Controller

Manufacturer DELL

Status OK

Appendix C – Tunable Parameters

PNP Device ID PCIIVEN_101E&DEV_1960&SUBSYS_04941028&REV_01\3&1070020&0&40
Memory Address 0xF8000000-0xFCFFFFFF
IRQ Channel IRQ 20
Driver c:\windows\system32\drivers\mraid35x.sys (5.2.22.4 built by: WinDDK, 17.88 KB (18,304 bytes), 1/9/2003 11:54 AM)

[IDE]

Item Value
Name CSB5 IDE Controller
Manufacturer ServerWorks
Status OK
PNP Device ID PCIIVEN_1166&DEV_0212&SUBSYS_02121166&REV_93\3&13C0B0C5&0&79
I/O Port 0x000008B0-0x000008BF
Driver c:\windows\system32\drivers\pciide.sys (5.2.3790.0 (srv03_rtm.030324-2048), 5.50 KB (5,632 bytes), 3/29/2003 12:00 AM)

Name Primary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI controllers)
Status OK
PNP Device ID PCIIDE\IDECHANNEL\4&10A8249&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), 3/29/2003 12:00 AM)

Name Secondary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI controllers)
Status OK
PNP Device ID PCIIDE\IDECHANNEL\4&10A8249&0&1
I/O Port 0x00000170-0x00000177
I/O Port 0x00000376-0x00000376
Driver c:\windows\system32\drivers\atapi.sys (5.2.3790.0 (srv03_rtm.030324-2048), 89.00 KB (91,136 bytes), 3/29/2003 12:00 AM)

[Printing]

Name	Driver	Port Name	Server Name
------	--------	-----------	-------------

[Problem Devices]

Device PNP Device ID Error Code
PCI Device
PCIIVEN_1028&DEV_000C&SUBSYS_000C1028&REV_00\3&13C0B0C5&0&20
The drivers for this device are not installed.
PCI Device PCIIVEN_1028&DEV_0008&SUBSYS_00081028&REV_00\3&13C0B0C5&0&21
The drivers for this device are not installed.
PCI Device
PCIIVEN_1028&DEV_000D&SUBSYS_000D1028&REV_00\3&13C0B0C5&0&22
The drivers for this device are not installed.

[USB]

Device PNP Device ID

Appendix C – Tunable Parameters

ServerWorks (RCC) PCI to USB Open Host Controller

PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_05\3&13C0B0C5&0&7A
 USB Root Hub USB\ROOT_HUB\4&1A0F8909&0

[Software Environment]

[System Drivers]

Name	Description	File	Type	Started	Start Mode	State	Status	Error Control
	Accept Pause	Accept Stop						
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	Kernel Driver	No				
	Disabled	Stopped	OK	Normal	No	No		
adpu160m	adpu160m	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
adpu320	adpu320	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
afcnt	afcnt	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
afd	AFD Networking Support Environment	c:\windows\system32\drivers\afd.sys	Kernel					
Driver	Yes	Auto	Running	OK	Normal	No	Yes	
aha154x	Aha154x	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
aic78u2	aic78u2	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
aic78xx	aic78xx	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
aliide	Aliide	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
asynmac	RAS Asynchronous Media Driver	c:\windows\system32\drivers\asynmac.sys	Kernel Driver	No	Manual			
	Stopped	OK	Normal	No	No			
atapi	Standard IDE/ESDI Hard Disk Controller	c:\windows\system32\drivers\atapi.sys	Kernel					
Driver	Yes	Boot	Running	OK	Normal	No	Yes	
atdisk	Atdisk	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Ignore	No	No					
ati2mpad	ati2mpad	c:\windows\system32\drivers\ati2mpad.sys	Kernel Driver					
	Yes	Manual	Running	OK	Ignore	No	Yes	
atmarpc	ATM ARP Client Protocol	c:\windows\system32\drivers\atmarpc.sys						
	Kernel Driver	No	Manual	Stopped	OK	Normal	No	No
audstub	Audio Stub Driver	c:\windows\system32\drivers\audstub.sys	Kernel Driver					
	Yes	Manual	Running	OK	Normal	No	Yes	
b57w2k	Broadcom NetXtreme Gigabit Ethernet	c:\windows\system32\drivers\b57xp32.sys						
	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes
beep	Beep	c:\windows\system32\drivers\beep.sys	Kernel Driver	Yes	System			
	Running	OK	Normal	No	Yes			
cbidf2k	cbidf2k	c:\windows\system32\drivers\cbidf2k.sys	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
cd20xrnt	cd20xrnt	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			

Appendix C – Tunable Parameters

cdfs	Cdfs	c:\windows\system32\drivers\cdfs.sys	File System Driver	Yes				
	Disabled	Running	OK	Normal	No	Yes		
cdrom	CD-ROM Driver	c:\windows\system32\drivers\cdrom.sys	Kernel Driver	Yes	System			
	Running	OK	Normal	No	Yes			
changer	Changer	Not Available	Kernel Driver	No	System Stopped			
	OK	Ignore	No					
clusdisk	Cluster Disk Driver	c:\windows\system32\drivers\clusdisk.sys	Kernel Driver					
	No	Disabled	Stopped	OK	Normal	No	No	
cmdide	CmdIde	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
cpqarray	Cpqarray	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
cpqarray2	cpqarray2	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
cpqcissm	cpqcissm	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
cpqcalm	cpqcalm	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
crdisk	CRC Disk Filter Driver	c:\windows\system32\drivers\crdisk.sys	Kernel Driver	Yes				
	Boot	Running	OK	Normal	No	Yes		
dac960nt	dac960nt	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
dcesmwdm	Instrumentation service device driver	c:\windows\system32\drivers\dcesmwdm.sys	Kernel Driver	No	Manual			
	Stopped	OK	Normal	No	No			
dellcerc	dellcerc	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
dfsdriver	DfsDriver	c:\windows\system32\drivers\dfs.sys	File System Driver					
	Yes	Boot	Running	OK	Normal	No	Yes	
disk	Disk Driver	c:\windows\system32\drivers\disk.sys	Kernel Driver	Yes	Boot			
	Running	OK	Normal	No	Yes			
dmboot	dmboot	c:\windows\system32\drivers\dmboot.sys	Kernel Driver	No				
	Disabled	Stopped	OK	Normal	No	No		
dmio	Logical Disk Manager Driver	c:\windows\system32\drivers\dmio.sys	Kernel Driver					
	Yes	Boot	Running	OK	Normal	No	Yes	
dmload	dmload	c:\windows\system32\drivers\dmload.sys	Kernel Driver	Yes	Boot			
	Running	OK	Normal	No	Yes			
dpti2o	dpti2o	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
fastfat	Fastfat	c:\windows\system32\drivers\fastfat.sys	File System Driver	Yes				
	Disabled	Running	OK	Normal	No	Yes		
fdc	Floppy Disk Controller Driver	c:\windows\system32\drivers\fdc.sys	Kernel Driver					
	Yes	Manual	Running	OK	Normal	No	Yes	
fips	Fips	c:\windows\system32\drivers\fips.sys	Kernel Driver	Yes	System			
	Running	OK	Normal	No	Yes			
flpydisk	Floppy Disk Driver	c:\windows\system32\drivers\flpydisk.sys	Kernel Driver					
	Yes	Manual	Running	OK	Normal	No	Yes	
ftdisk	Volume Manager Driver	c:\windows\system32\drivers\ftdisk.sys	Kernel Driver	Yes				
	Boot	Running	OK	Normal	No	Yes		
gpc	Generic Packet Classifier	c:\windows\system32\drivers\msgpc.sys	Kernel Driver					
	Yes	Manual	Running	OK	Normal	No	Yes	
hpn	hpn	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					
hpt3xx	hpt3xx	Not Available	Kernel Driver	No	Disabled	Stopped		OK
	Normal	No	No					

Appendix C – Tunable Parameters

http	HTTP	c:\windows\system32\drivers\http.sys	Kernel Driver	No	Manual		
	Stopped	OK	Normal	No	No		
i2omgmt	i2omgmt	Not Available	Kernel Driver	No	System Stopped		
	OK	Normal	No	No			
i2omp	i2omp	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No	No				
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver						
		c:\windows\system32\drivers\i8042prt.sys	Kernel Driver	Yes	System		
	Running	OK	Normal	No	Yes		
iirsp	iirsp	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No	No				
imapi	CD-Burning Filter Driver	c:\windows\system32\drivers\imapi.sys	Kernel Driver	No			
	System Stopped	OK	Normal	No	No		
intelide	IntelIde	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No	No				
ipfilterdriver	IP Traffic Filter Driver	c:\windows\system32\drivers\ipfltdrv.sys	Kernel Driver				
	No	Manual Stopped	OK	Normal	No	No	
ipinip	IP in IP Tunnel Driver	c:\windows\system32\drivers\ipinip.sys	Kernel Driver	No			
	Manual Stopped	OK	Normal	No	No		
ipnat	IP Network Address Translator	c:\windows\system32\drivers\ipnat.sys	Kernel Driver				
	No	Manual Stopped	OK	Normal	No	No	
ipsec	IPSEC driver	c:\windows\system32\drivers\ipsec.sys	Kernel Driver	Yes	System		
	Running	OK	Normal	No	Yes		
ipsraidn	ipsraidn	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No	No				
irenum	IR Enumerator Service	c:\windows\system32\drivers\irenum.sys	Kernel Driver	No			
	Manual Stopped	OK	Normal	No	No		
isapnp	PnP ISA/EISA Bus Driver	c:\windows\system32\drivers\isapnp.sys	Kernel Driver				
	Yes	Boot Running	OK	Critical	No	Yes	
kbdclass	Keyboard Class Driver	c:\windows\system32\drivers\kbdclass.sys	Kernel				
Driver	Yes	System Running	OK	Normal	No	Yes	
ksecdd	KSecDD	c:\windows\system32\drivers\ksecdd.sys	Kernel Driver	Yes	Boot		
	Running	OK	Normal	No	Yes		
lp6nds35	lp6nds35	Not Available	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No		
mnmdd	mnmdd	c:\windows\system32\drivers\mnmdd.sys	Kernel Driver	Yes	System		
	Running	OK	Ignore	No	Yes		
modem	Modem	c:\windows\system32\drivers\modem.sys	Kernel Driver	No	Manual		
	Stopped	OK	Ignore	No	No		
mouclass	Mouse Class Driver	c:\windows\system32\drivers\mouclass.sys	Kernel				
Driver	Yes	System Running	OK	Normal	No	Yes	
mountmgr	Mount Point Manager	c:\windows\system32\drivers\mountmgr.sys	Kernel				
Driver	Yes	Boot Running	OK	Normal	No	Yes	
mraid35x	mraid35x	c:\windows\system32\drivers\mraid35x.sys	Kernel Driver				
	Yes	Boot Running	OK	Normal	No	Yes	
mrxdav	WebDav Client Redirector	c:\windows\system32\drivers\mrxdav.sys	File				
System Driver	No	Manual Stopped	OK	Normal	No	No	
mrxsmbr	MRXSMB	c:\windows\system32\drivers\mrxsmbr.sys	File System Driver				
	Yes	System Running	OK	Normal	No	Yes	
msfs	Msfs	c:\windows\system32\drivers\msfs.sys	File System Driver	Yes	System		
	Running	OK	Normal	No	Yes		
mup	Mup	c:\windows\system32\drivers\mup.sys	File System Driver	Yes	Boot		
	Running	OK	Normal	No	Yes		
ndis	NDIS System Driver	c:\windows\system32\drivers\ndis.sys	Kernel Driver	Yes			
	Boot Running	OK	Normal	No	Yes		

Appendix C – Tunable Parameters

ndistapi	Remote Access NDIS TAPI Driver	c:\windows\system32\drivers\ndistapi.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes		
ndisuio	NDIS Usermode I/O Protocol	c:\windows\system32\drivers\ndisuio.sys	Kernel Driver	No	Manual	Stopped	OK	Normal	No	No		
ndiswan	Remote Access NDIS WAN Driver	c:\windows\system32\drivers\ndiswan.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes		
ndproxy	NDIS Proxy	c:\windows\system32\drivers\ndproxy.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes		
netbios	NetBIOS Interface	c:\windows\system32\drivers\netbios.sys	File System	Driver	Yes	System	Running	OK	Normal	No	Yes	
netbt	NetBios over Tcpip	c:\windows\system32\drivers\netbt.sys	Kernel Driver	Yes	System	Running	OK	Normal	No	Yes		
nfrd960	nfrd960	Not Available	Kernel Driver	No	Manual	Stopped	OK	Disabled				
npfs	Npfs	c:\windows\system32\drivers\npfs.sys	File System	Driver	Yes	System	Running	OK	Normal	No	Yes	
ntfs	Ntfs	c:\windows\system32\drivers\ntfs.sys	File System	Driver	Yes	System	Disabled	Running	OK	Normal	No	Yes
null	Null	c:\windows\system32\drivers\null.sys	Kernel Driver	Yes	System	Running	OK	Normal	No	Yes		
parport	Parport	c:\windows\system32\drivers\parport.sys	Kernel Driver	No	Manual	Stopped	OK	Ignore	No	No		
partmgr	Partition Manager	c:\windows\system32\drivers\partmgr.sys	Kernel Driver	Yes	Boot	Running	OK	Normal	No	Yes		
pci	PCI Bus Driver	c:\windows\system32\drivers\pci.sys	Kernel Driver	Yes	Boot	Running	OK	Critical	No	Yes		
pciide	PCIIde	c:\windows\system32\drivers\pciide.sys	Kernel Driver	Yes	Boot	Running	OK	Normal	No	Yes		
pcmcia	Pcmcia	c:\windows\system32\drivers\pcmcia.sys	Kernel Driver	No	Manual	Stopped	OK	Normal	No	No		
pdcomp	PDCOMP	Not Available	Kernel Driver	No	Manual	Stopped	OK	Ignore	No	No		
pdframe	PDFRAME	Not Available	Kernel Driver	No	Manual	Stopped	OK	Ignore	No	No		
pdreli	PDRELI	Not Available	Kernel Driver	No	Manual	Stopped	OK	Ignore	No	No		
pdrframe	PDRFRAME	Not Available	Kernel Driver	No	Manual	Stopped	OK	Ignore	No	No		
perc2	perc2	c:\windows\system32\drivers\perc2.sys	Kernel Driver	Yes	Boot	Running	OK	Normal	No	Yes		
perc2hib	perc2hib	c:\windows\system32\drivers\perc2hib.sys	Kernel Driver	Yes	Boot	Running	OK	Normal	No	Yes		
pptpminiport	WAN Miniport (PPTP)	c:\windows\system32\drivers\raspptp.sys	Kernel	Driver	Yes	Manual	Running	OK	Normal	No	Yes	
processor	Processor Driver	c:\windows\system32\drivers\processr.sys	Kernel	Driver	Yes	Manual	Running	OK	Normal	No	Yes	
ptilink	Direct Parallel Link Driver	c:\windows\system32\drivers\ptilink.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes		
ql1080	ql1080	Not Available	Kernel Driver	No	Manual	Stopped	OK	Disabled				
ql10wnt	Ql10wnt	Not Available	Kernel Driver	No	Manual	Stopped	OK	Disabled				
ql12160	ql12160	Not Available	Kernel Driver	No	Manual	Stopped	OK	Disabled				

Appendix C – Tunable Parameters

ql1240	ql1240	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
ql1280	ql1280	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
ql2100	ql2100	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
ql2200	ql2200	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
ql2300	ql2300	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
rasacd	Remote Access Auto Connection Driver		c:\windows\system32\drivers\rasacd.sys	Kernel Driver	Yes	System Running	OK
	Driver	Yes	System Running	OK	Normal	No	Yes
rasl2tp	WAN Miniport (L2TP)		c:\windows\system32\drivers\rasl2tp.sys	Kernel Driver	Yes	Manual Running	OK
	Manual Running	OK	Normal	No	Yes		
rasppoe	Remote Access PPPOE Driver		c:\windows\system32\drivers\rasppoe.sys	Kernel Driver	Yes	Manual Running	OK
	Kernel Driver	Yes	Manual Running	OK	Normal	No	Yes
raspti	Direct Parallel		c:\windows\system32\drivers\raspti.sys	Kernel Driver	Yes	Running	OK
	Running	OK	Normal	No	Yes		
rdcss	Rdbss		c:\windows\system32\drivers\rdbss.sys	File System Driver	Yes	Running	OK
	Running	OK	Normal	No	Yes		
rdpcdd	RDPcdd		c:\windows\system32\drivers\rdpcdd.sys	Kernel Driver	Yes	Running	OK
	Running	OK	Ignore	No	Yes		
rdpdr	Terminal Server Device Redirector Driver		c:\windows\system32\drivers\rdpdr.sys	Kernel Driver	Yes	Manual Running	OK
	Kernel Driver	Yes	Manual Running	OK	Normal	No	Yes
rdpwd	RDPWD		c:\windows\system32\drivers\rdpwd.sys	Kernel Driver	Yes	Running	OK
	Running	OK	Ignore	No	Yes		
redbook	Digital CD Audio Playback Filter Driver		c:\windows\system32\drivers\redbook.sys	Kernel Driver	Yes	Running	OK
	Running	OK	Normal	No	Yes		
scsiprnt	Microsoft SCSI/1394 Generic Printer Class		c:\windows\system32\drivers\scsiprnt.sys	Kernel Driver	No	Stopped	OK
	Stopped	OK	Normal	No	No		
secdrv	Secdrv		c:\windows\system32\drivers\secdrv.sys	Kernel Driver	No	Stopped	OK
	Stopped	OK	Normal	No	No		
serenum	Serenum Filter Driver		c:\windows\system32\drivers\serenum.sys	Kernel Driver	Yes	Manual Running	OK
	Driver	Yes	Manual Running	OK	Normal	No	Yes
serial	Serial port driver		c:\windows\system32\drivers\serial.sys	Kernel Driver	Yes	System Running	OK
	System Running	OK	Ignore	No	Yes		
sfloppy	Sfloppy		c:\windows\system32\drivers\sfloppy.sys	Kernel Driver	No	Stopped	OK
	Stopped	OK	Ignore	No	No		
simbad	Simbad	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
sparrow	Sparrow	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	OK	Normal	No	No			
srv	Srv		c:\windows\system32\drivers\srv.sys	File System Driver	Yes	Running	OK
	Running	OK	Normal	No	Yes		
swenum	Software Bus Driver		c:\windows\system32\drivers\swenum.sys	Kernel Driver	Yes	Manual Running	OK
	Driver	Yes	Manual Running	OK	Normal	No	Yes
symc810	symc810	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Stopped	OK	Normal	No	No		
symc8xx	symc8xx	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Stopped	OK	Normal	No	No		
symmpi	symmpi	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					

Appendix C – Tunable Parameters

sym_hi	sym_hi	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
sym_u3	sym_u3	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
tcpip	TCP/IP Protocol Driver		c:\windows\system32\drivers\tcpip.sys	Kernel Driver	Yes		
	System Running	OK	Normal	No	Yes		
tdpipe	TDPIPE		c:\windows\system32\drivers\tdpipe.sys	Kernel Driver	No	Manual	
	Stopped	OK	Ignore	No	No		
tdtcp	TDTCP		c:\windows\system32\drivers\tdtcp.sys	Kernel Driver	Yes	Manual	
	Running	OK	Ignore	No	Yes		
termdd	Terminal Device Driver		c:\windows\system32\drivers\termdd.sys	Kernel Driver	Yes		
	System Running	OK	Normal	No	Yes		
toside	Toside	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
udfs	Udfs		c:\windows\system32\drivers\udfs.sys	File System Driver	No		
	Disabled	Stopped	OK	Normal	No	No	
ultra	ultra	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
update	Microcode Update Driver		c:\windows\system32\drivers\update.sys	Kernel Driver			
	Yes	Manual	Running	OK	Normal	No	Yes
usbhub	USB2 Enabled Hub		c:\windows\system32\drivers\usbhub.sys	Kernel Driver			
	Yes	Manual	Running	OK	Normal	No	Yes
usbohci	Microsoft USB Open Host Controller Miniport Driver		c:\windows\system32\drivers\usbohci.sys	Kernel Driver	Yes	Manual	
	Running	OK	Normal	No	Yes		
usbstor	USB Mass Storage Driver		c:\windows\system32\drivers\usbstor.sys	Kernel			
	Driver	No	Manual	Stopped	OK	Normal	No
vgasave	VGA Display Controller		c:\windows\system32\drivers\vga.sys	Kernel Driver			
	Yes	System	Running	OK	Ignore	No	Yes
viaide	Vialde	Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No					
volsnap	Storage volumes		c:\windows\system32\drivers\volsnap.sys	Kernel Driver			
	Yes	Boot	Running	OK	Normal	No	Yes
wanarp	Remote Access IP ARP Driver		c:\windows\system32\drivers\wanarp.sys	Kernel			
	Driver	Yes	Manual	Running	OK	Normal	No
wdica	WDICA	Not Available	Kernel Driver	No	Manual	Stopped	OK
	No	No					Ignore
wlbs	Network Load Balancing		c:\windows\system32\drivers\wlbs.sys	Kernel Driver			
	No	Manual	Stopped	OK	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	Not Available	Not 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	No	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor	
types)	cpu.inf	Not Available	ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\0			
Processor	No	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor	
types)	cpu.inf	Not Available	ACPI\GENUINEINTEL_-_X86_FAMILY_15_MODEL_2\2			

Appendix C – Tunable Parameters

PCI busNo	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	
machine.inf	Not Available	ACPI\PNP0A03\1			
ServerWorks (RCC) CMIC_LE Processor to PCI Bridge(*)	No	SYSTEM			
5.2.3790.0	10/1/2002	ServerWorks (RCC)	machine.inf	Not Available	
PCIIVEN_1166&DEV_0014&SUBSYS_00000000&REV_32\3&13C0B0C5&0&00					
ServerWorks (RCC) CMIC_LE Processor to PCI Bridge(*)	No	SYSTEM			
5.2.3790.0	10/1/2002	ServerWorks (RCC)	machine.inf	Not Available	
PCIIVEN_1166&DEV_0014&SUBSYS_00000000&REV_00\3&13C0B0C5&0&01					
ServerWorks (RCC) CMIC_LE Processor to PCI Bridge(*)	No	SYSTEM			
5.2.3790.0	10/1/2002	ServerWorks (RCC)	machine.inf	Not Available	
PCIIVEN_1166&DEV_0014&SUBSYS_00000000&REV_00\3&13C0B0C5&0&02					
PCI Device	Not Available	UNKNOWN	Not Available	Not Available	Not Available
Not Available	Not Available				
PCIIVEN_1028&DEV_000C&SUBSYS_000C1028&REV_00\3&13C0B0C5&0&20					
PCI Device	Not Available	UNKNOWN	Not Available	Not Available	Not Available
Not Available	Not Available				
PCIIVEN_1028&DEV_0008&SUBSYS_00081028&REV_00\3&13C0B0C5&0&21					
PCI Device	Not Available	UNKNOWN	Not Available	Not Available	Not Available
Not Available	Not Available				
PCIIVEN_1028&DEV_000D&SUBSYS_000D1028&REV_00\3&13C0B0C5&0&22					
RAGE XL PCI Family (Microsoft Corporation)	No	DISPLAY	5.10.2600.6014		
8/8/2001	ATI Technologies Inc.	atiixpad.inf	Not Available		
PCIIVEN_1002&DEV_4752&SUBSYS_01211028&REV_27\3&13C0B0C5&0&70					
Default Monitor	No	MONITOR	5.1.2001.0	6/6/2001	(Standard monitor types)
monitor.inf	Not Available				
DISPLAY\DEFAULT_MONITOR\4&2664298A&0&80000000&00&0E					
ServerWorks Champion CSB5 - SouthBridge 5	No	SYSTEM	5.2.3790.0		
10/1/2002	ServerWorks (RCC)	machine.inf	Not Available		
PCIIVEN_1166&DEV_0201&SUBSYS_00000000&REV_93\3&13C0B0C5&0&78					
Direct memory access controller	No	SYSTEM	5.2.3790.0	10/1/2002	
(Standard system devices)		machine.inf	Not Available		
ACPI\PNP0200\4&25F73A82&0					
Numeric data processor	No	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
machine.inf	Not Available	ACPI\PNP0C04\4&25F73A82&0			
Programmable interrupt controller	No	SYSTEM	5.2.3790.0	10/1/2002	
(Standard system devices)		machine.inf	Not Available		
ACPI\PNP0000\4&25F73A82&0					
System speaker	No	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
machine.inf	Not Available	ACPI\PNP0800\4&25F73A82&0			
System timer	No	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
machine.inf	Not Available	ACPI\PNP0100\4&25F73A82&0			
Standard floppy disk controller	No	FDC	5.2.3790.0	10/1/2002	(Standard floppy disk controllers)
fdc.inf	Not Available	ACPI\PNP0700\4&25F73A82&0			
Floppy disk drive	No	FLOPPYDISK	5.2.3790.0	10/1/2002	(Standard floppy disk drives)
flpydisk.inf	Not Available				
FDC\GENERIC_FLOPPY_DRIVE\5&1AE2F47D&0&0					
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\4&25F73A82&0					
PS/2 Compatible Mouse	No	MOUSE	5.2.3790.0	10/1/2002	Microsoft
msmouse.inf	Not Available	ACPI\PNP0F13\4&25F73A82&0			
Communications Port	No	PORTS	5.2.3790.0	10/1/2002	(Standard port types)
msports.inf	Not Available	ACPI\PNP0501\1			
Communications Port	No	PORTS	5.2.3790.0	10/1/2002	(Standard port types)
msports.inf	Not Available	ACPI\PNP0501\2			

Appendix C – Tunable Parameters

System CMOS/real time clock No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 ACPI\PNP0B00\4&25F73A82&0

System board No SYSTEM 5.2.3790.0 10/1/2002 (Standard system
 devices) machine.inf Not Available ACPI\PNP0C01\2
 CSB5 IDE Controller No HDC 5.2.3790.0 10/1/2002 ServerWorks
 mshdc.inf Not Available
 PCI\VEN_1166&DEV_0212&SUBSYS_02121166&REV_93\3&13C0B0C5&0&79

Primary IDE Channel No HDC 5.2.3790.0 10/1/2002 (Standard IDE
 ATA/ATAPI controllers) mshdc.inf Not Available
 PCI\IDE\IDECHANNEL\4&10A8249&0&0

CD-ROM Drive No CDROM 5.2.3790.0 10/1/2002 (Standard CD-ROM
 drives) cdrom.inf Not Available IDE\CDROMSAMSUNG_CD-ROM_SN-
 124_____Q009____\5&3125DC91&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
 PCI\IDE\IDECHANNEL\4&10A8249&0&1

ServerWorks (RCC) PCI to USB Open Host Controller No USB 5.2.3790.0
 10/1/2002 ServerWorks (RCC) usbport.inf Not Available
 PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_05\3&13C0B0C5&0&7A

USB Root Hub No USB 5.2.3790.0 10/1/2002 (Standard USB Host Controller)
 usbport.inf Not Available USB\ROOT_HUB\4&1A0F8909&0

Serverworks Champion CSB5 - SouthBridge 5 LPC No SYSTEM 5.2.3790.0
 10/1/2002 ServerWorks (RCC) machine.inf Not Available
 PCI\VEN_1166&DEV_0225&SUBSYS_00000000&REV_00\3&13C0B0C5&0&7B

ISAPNP Read Data Port No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 ISAPNP\READDATAPORT\0

ServerWorks Grand Champion CIOB_X2 - I/O Bridge 133 Mhz No SYSTEM
 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf Not Available
 PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV_05\3&13C0B0C5&0&80

ServerWorks Grand Champion CIOB_X2 - I/O Bridge 133 Mhz No SYSTEM
 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf Not Available
 PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV_05\3&13C0B0C5&0&82

ServerWorks Grand Champion CIOB_X2 - I/O Bridge 133 Mhz No SYSTEM
 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf Not Available
 PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV_03\3&13C0B0C5&0&88

ServerWorks Grand Champion CIOB_X2 - I/O Bridge 133 Mhz No SYSTEM
 5.2.3790.0 10/1/2002 ServerWorks (RCC) machine.inf Not Available
 PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV_03\3&13C0B0C5&0&8A

PCI bus No SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPI\PNP0A03\5

PCI standard PCI-to-PCI bridge No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 PCI\VEN_8086&DEV_0309&SUBSYS_00000000&REV_01\3&474B838&0&40

Dell PERC 3 RAID (SCSI chip) No SCSIADAPTER 5.2.3790.0 10/1/2002 Dell
 pnpscsi.inf Not Available
 PCI\VEN_9005&DEV_00C5&SUBSYS_00C51028&REV_01\4&22300438&0&3040

Dell PERC 3 RAID (SCSI chip) No SCSIADAPTER 5.2.3790.0 10/1/2002 Dell
 pnpscsi.inf Not Available
 PCI\VEN_9005&DEV_00C5&SUBSYS_00C51028&REV_01\4&22300438&0&3140

Dell PERC 3/Di RAID Controller No SCSIADAPTER 5.2.3790.0 10/1/2002 Dell
 pnpscsi.inf Not Available
 PCI\VEN_1028&DEV_000A&SUBSYS_01211028&REV_01\3&474B838&0&41

Appendix C – Tunable Parameters

Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk drives)
 disk.inf Not Available
 SCSIIDISK&VEN_DELL&PROD_CONTAINER&REV_V1.0\4&318925A3&0&400

Dell PERC 2 Management Device No SYSTEM 5.2.3790.0 10/1/2002
 Adaptec scsidev.inf Not Available
 SCSIIPROCESSOR&VEN_DELL&PROD_MANAGEMENT&REV_V1.0\4&318925A3&0&500

PCI busNo SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPIPNP0A03\4

Broadcom NetXtreme Gigabit Ethernet No NET 2.91.0.0 10/1/2002
 Broadcom netb57xp.inf Not Available
 PCIIVEN_14E4&DEV_16A7&SUBSYS_01211028&REV_02\3&172E68DD&0&30

Broadcom NetXtreme Gigabit Ethernet No NET 2.91.0.0 10/1/2002
 Broadcom netb57xp.inf Not Available
 PCIIVEN_14E4&DEV_16A7&SUBSYS_01211028&REV_02\3&172E68DD&0&40

PCI busNo SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPIPNP0A03\3

DELL PERC 3/DC Plus RAID Controller No SCSIADAPTER 5.2.22.4 12/3/2002
 DELL oem1.inf Not Available
 PCIIVEN_101E&DEV_1960&SUBSYS_04941028&REV_01\3&29E81982&0&30

DELL PV22XS Backplane No SYSTEM 5.2.3790.0 10/1/2002 Dell
 scsidev.inf Not Available
 SCSIIPROCESSOR&VEN_DELL&PROD_PV22XS&REV_E.10\4&19309C39&0&060

DELL PV22XS Backplane No SYSTEM 5.2.3790.0 10/1/2002 Dell
 scsidev.inf Not Available
 SCSIIPROCESSOR&VEN_DELL&PROD_PV22XS&REV_E.10\4&19309C39&0&160

RAID Virtual Device No SYSTEM 5.2.3790.0 10/1/2002 American
 Megatrends, Inc. scsidev.inf Not Available
 SCSIOTHER&VEN__RAID&PROD_DUMMYDEVICE&REV_0000\4&19309C39&0&2

F0

Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk drives)
 disk.inf Not Available
 SCSIIDISK&VEN_PERC&PROD_LD_0_PERCRAID&REV_4&19309C39&0&300

PCI busNo SYSTEM 5.2.3790.0 10/1/2002 (Standard system devices)
 machine.inf Not Available ACPIPNP0A03\2

DELL PERC 3/DC Plus RAID Controller No SCSIADAPTER 5.2.22.4 12/3/2002
 DELL oem1.inf Not Available
 PCIIVEN_101E&DEV_1960&SUBSYS_04941028&REV_01\3&1070020&0&40

DELL PV22XS Backplane No SYSTEM 5.2.3790.0 10/1/2002 Dell
 scsidev.inf Not Available
 SCSIIPROCESSOR&VEN_DELL&PROD_PV22XS&REV_E.10\4&116608EE&0&060

DELL PV22XS Backplane No SYSTEM 5.2.3790.0 10/1/2002 Dell
 scsidev.inf Not Available
 SCSIIPROCESSOR&VEN_DELL&PROD_PV22XS&REV_E.10\4&116608EE&0&160

RAID Virtual Device No SYSTEM 5.2.3790.0 10/1/2002 American
 Megatrends, Inc. scsidev.inf Not Available
 SCSIOTHER&VEN__RAID&PROD_DUMMYDEVICE&REV_0000\4&116608EE&0&2

F0

Disk drive No DISKDRIVE 5.2.3790.0 10/1/2002 (Standard disk drives)
 disk.inf Not Available
 SCSIIDISK&VEN_PERC&PROD_LD_0_PERCRAID&REV_4&116608EE&0&300

ACPI Fixed Feature Button No SYSTEM 5.2.3790.0 10/1/2002
 (Standard system devices) machine.inf Not Available
 ACPIFIXEDBUTTON\2&DABA3FF&0

Appendix C – Tunable Parameters

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&SIGNATURECF72CF72OFFSET7E00LENGTH23					
254F800					
Generic volume No	VOLUME		5.2.3790.0	10/1/2002	Microsoft
volume.inf		Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATURECF72CF72OFFSET232557600LEN					
GTHEC082AE00					
Generic volume No	VOLUME		5.2.3790.0	10/1/2002	Microsoft
volume.inf		Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREEE18FCF8BOFFSET7E00LENGTH27					
106BDE00					
Generic volume No	VOLUME		5.2.3790.0	10/1/2002	Microsoft
volume.inf		Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREEE18FCF8BOFFSET27106C5C00LE					
NGTH27106C5C00					
Generic volume No	VOLUME		5.2.3790.0	10/1/2002	Microsoft
volume.inf		Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREEE18FCF8BOFFSET4E20D8B800LE					
NGTH27FB429A00					
Generic volume No	VOLUME		5.2.3790.0	10/1/2002	Microsoft
volume.inf		Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREEE18FCF85OFFSET7E00LENGTH27					
106BDE00					
Generic volume No	VOLUME		5.2.3790.0	10/1/2002	Microsoft
volume.inf		Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREEE18FCF85OFFSET27106C5C00LE					
NGTH27106C5C00					
Generic volume No	VOLUME		5.2.3790.0	10/1/2002	Microsoft
volume.inf		Not Available			
STORAGE\VOLUME\1&30A96598&0&SIGNATUREEE18FCF85OFFSET4E20D8B800LE					
NGTH27FB429A00					
AFD Networking Support Environment		Not Available	LEGACYDRIVER		Not Available
Not Available		Not Available	Not Available		
ROOT\LEGACY_AFD\0000					
Beep	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available		Not Available	ROOT\LEGACY_BEEP\0000		
CRC Disk Filter Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available		Not Available	ROOT\LEGACY_CRCDISK\0000		
dmbboot	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available		Not Available	ROOT\LEGACY_DMBOOT\0000		
dmload	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available		Not Available	ROOT\LEGACY_DMLOAD\0000		
Fips	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available		Not Available	ROOT\LEGACY_FIPS\0000		
Generic Packet Classifier		Not Available	LEGACYDRIVER	Not Available	Not
Available		Not Available	Not Available	ROOT\LEGACY_GPC\0000	
IPSEC driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not
Available		Not Available	ROOT\LEGACY_IPSEC\0000		
ksecdd	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available		Not Available	ROOT\LEGACY_KSECDD\0000		

Appendix C – Tunable Parameters

mnmdd	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_MNMDD\0000		
mountmgr	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_MOUNTMGR\0000		
NDIS System Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	
	Not Available	Not Available	ROOTLEGACY_NDIS\0000		
Remote Access NDIS TAPI Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	
	Not Available	Not Available	ROOTLEGACY_NDISTAPI\0000		
NDIS Usermode I/O Protocol	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_NDISUIO\0000		
NDProxy	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_NDPROXY\0000		
NetBios over Tcip	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_NETBT\0000		
Null	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_NULL\0000		
Partition Manager	Not Available	LEGACYDRIVER	Not Available	Not Available	
	Not Available	Not Available	ROOTLEGACY_PARTMGR\0000		
Remote Access Auto Connection Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	
	Not Available	Not Available	ROOTLEGACY_RASACD\0000		
RDPCDD	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_RDPCDD\0000		
RDPWD	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_RDPWD\0000		
TCP/IP Protocol Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	
	Not Available	Not Available	ROOTLEGACY_TCPIP\0000		
TDPIPE	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_TDPIPE\0000		
TDTCP	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_TDTCP\0000		
VGA Display Controller.	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_VGASAVE\0000		
volsnap	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_VOLSNAP\0000		
Remote Access IP ARP Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOTLEGACY_WANARP\0000		
Audio Codecs	No	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
	wave.inf	Not Available	ROOTMEDIA\MS_MMACM		
Legacy Audio Drivers	No	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
	wave.inf	Not Available	ROOTMEDIA\MS_MMDRV		
Media Control Devices	No	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
	wave.inf	Not Available	ROOTMEDIA\MS_MMMCI		
Legacy Video Capture Devices	No	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
	wave.inf	Not Available	ROOTMEDIA\MS_MMVCD		
Video Codecs	No	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
	wave.inf	Not Available	ROOTMEDIA\MS_MMVID		
WAN Miniport (L2TP)	No	NET 5.2.3790.0	10/1/2002	Microsoft	
	netrasa.inf	Not Available	ROOTMS_L2TPMINIPORT\0000		
WAN Miniport (IP)	No	NET 5.2.3790.0	10/1/2002	Microsoft	
	netrasa.inf	Not Available	ROOTMS_NDISWANIP\0000		

Appendix C – Tunable Parameters

WAN Miniport (PPPOE) No	NET	5.2.3790.0	10/1/2002	Microsoft	
netrasa.inf	Not Available	ROOT\MS_PPPOEMINIPOINT\0000			
WAN Miniport (PPTP) No	NET	5.2.3790.0	10/1/2002	Microsoft	
netrasa.inf	Not Available	ROOT\MS_PPTPMINIPOINT\0000			
Direct Parallel No	NET	5.2.3790.0	10/1/2002	Microsoft	netrasa.inf
Not Available		ROOT\MS_PTMINIPOINT\0000			
Terminal Server Device Redirector (Standard system devices)	No	SYSTEM	5.2.3790.0	10/1/2002	
		machine.inf	Not Available	ROOT\RDPPDR\0000	
Terminal Server Keyboard Driver (Standard system devices)	No	SYSTEM	5.2.3790.0	10/1/2002	
		machine.inf	Not Available	ROOT\RDPPDR_KBD\0000	
Terminal Server Mouse Driver (Standard system devices)	No	SYSTEM	5.2.3790.0	10/1/2002	
		machine.inf	Not Available	ROOT\RDPPDR_MOU\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 (Standard system devices)	No	SYSTEM	5.2.3790.0	10/1/2002	
		machine.inf	Not Available	ROOT\SYSTEM\0001	

[Environment Variables]

Variable	Value	User Name
ComSpec	%SystemRoot%\system32\cmd.exe	<SYSTEM>
Path	C:\MKS\mksnt;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\PROGRA~1\MICROS~1\80\Tools\BINN;C:\Program Files\Microsoft SQL Server\80\Tools\BINN	<SYSTEM>
windir	%SystemRoot%	<SYSTEM>
OS	Windows_NT	<SYSTEM>
PROCESSOR_ARCHITECTURE	x86	<SYSTEM>
PROCESSOR_LEVEL	15	<SYSTEM>
PROCESSOR_IDENTIFIER	x86 Family 15 Model 2 Stepping 5, GenuineIntel	<SYSTEM>
PROCESSOR_REVISION	0205	<SYSTEM>
NUMBER_OF_PROCESSORS	2	<SYSTEM>
ClusterLog	C:\WINDOWS\Cluster\cluster.log	<SYSTEM>
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH	<SYSTEM>
TEMP	%SystemRoot%\TEMP	<SYSTEM>
TMP	%SystemRoot%\TEMP	<SYSTEM>
ROOTDIR	C:\MKS	<SYSTEM>
SHELL	C:/MKS/mksnt/sh.exe	<SYSTEM>
HOME	C:/Documents and Settings/Administrator	<SYSTEM>
TMPDIR	C:/WINDOWS/TEMP	<SYSTEM>
TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
TEMP	%USERPROFILE%\Local Settings\Temp	PE2650\Administrator
TMP	%USERPROFILE%\Local Settings\Temp	PE2650\Administrator

[Print Jobs]

Document	Size	Owner	Notify	Status	Time Submitted	Start Time	Until Time
Elapsed Time	Pages	Printed	Job ID	Priority	Parameters	Driver	Print Processor
Host	Print Queue	Data Type	Name				

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
Dell			205	October 2004
TPC-C Full Disclosure Report				
Copyright Dell				

Appendix C – Tunable Parameters

[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	Not Available	Not Available
system	Not Available	4	8	0	1413120	Not Available	Not Available	Not Available	Not Available
smss.exe	Not Available	344	11	204800	1413120	1/8/2004 4:54 PM	Not Available	Not Available	Not Available
csrss.exe	Not Available	540	13	Not Available	Not Available	1/8/2004 4:54 PM	Not Available	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	564	13	204800	1413120	1/8/2004 4:54 PM	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/29/2003 12:00 AM
services.exe	c:\windows\system32\services.exe	608	9	204800	1413120	1/8/2004 4:54 PM	5.2.3790.0 (srv03_rtm.030324-2048)	102.00 KB (104,448 bytes)	3/29/2003 12:00 AM
lsass.exe	c:\windows\system32\lsass.exe	620	9	204800	1413120	1/8/2004 4:54 PM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
svchost.exe	c:\windows\system32\svchost.exe	800	8	204800	1413120	1/8/2004 4:54 PM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
svchost.exe	c:\windows\system32\svchost.exe	864	8	204800	1413120	1/8/2004 4:54 PM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
svchost.exe	c:\windows\system32\svchost.exe	1016	8	204800	1413120	1/8/2004 4:54 PM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
explorer.exe	c:\windows\explorer.exe	1188	8	204800	1413120	1/8/2004 4:55 PM	6.00.3790.0 (srv03_rtm.030324-2048)	1,008.50 KB (1,032,704 bytes)	3/29/2003 12:00 AM
sqlmangr.exe	c:\program files\microsoft sql server\80\tools\binn\sqlmangr.exe	1268	8	204800	1413120	1/8/2004 4:55 PM	2000.080.0760.00	72.57 KB (74,308 bytes)	11/7/2003 2:42 PM
winvnc.exe	c:\program files\orl\vnc\winvnc.exe	1276	8	204800	1413120	1/8/2004 4:55 PM	3, 3, 3, 7	204.00 KB (208,896 bytes)	5/30/2003 4:28 PM
mmc.exe	c:\windows\system32\mmc.exe	1424	8	204800	1413120	1/8/2004 4:55 PM	5.2.3790.0 (srv03_rtm.030324-2048)	762.50 KB (780,800 bytes)	3/29/2003 12:00 AM
cmd.exe	c:\windows\system32\cmd.exe	1828	8	204800	1413120	1/8/2004 4:55 PM	5.2.3790.0 (srv03_rtm.030324-2048)	374.00 KB (382,976 bytes)	3/29/2003 12:00 AM
sqlservr.exe	c:\program files\microsoft sql server\mssql\binn\sqlservr.exe	1872	13	204800	1413120	1/8/2004 4:55 PM	2000.080.0857.00	7.18 MB (7,532,584 bytes)	11/7/2003 2:42 PM
wmiprvse.exe	Not Available	276	8	Not Available	Not Available	1/8/2004 4:56 PM	Not Available	Not Available	Not Available
cmd.exe	c:\windows\system32\cmd.exe	1584	8	204800	1413120	1/9/2004 2:41 PM	5.2.3790.0 (srv03_rtm.030324-2048)	374.00 KB (382,976 bytes)	3/29/2003 12:00 AM

Appendix C – Tunable Parameters

```

isql.exe c:\progra~1\microso~1\80\tools\bin\isql.exe      1116  8      204800 1413120
          1/9/2004 2:41 PM      2000.080.0194.00      96.00 KB (98,304 bytes)11/7/2003 2:42
PM
tail.exe c:\mks\mksnt\tail.exe      200    8      204800 1413120      1/9/2004 2:42 PM
          5.2 build 63      43.50 KB (44,544 bytes)6/2/2003 11:09 AM
helpctr.exe c:\windows\pchealth\helpctr\binaries\helpctr.exe 180    8      204800
          1413120      1/9/2004 2:52 PM      5.2.3790.0 (srv03_rtm.030324-2048) 764.00
KB (782,336 bytes)      5/30/2003 3:10 PM
helpsvc.exe c:\windows\pchealth\helpctr\binaries\helpsvc.exe920    8      204800
          1413120      1/9/2004 2:52 PM      5.2.3790.0 (srv03_rtm.030324-2048) 720.00
KB (737,280 bytes)      5/30/2003 3:10 PM
helphost.exe c:\windows\pchealth\helpctr\binaries\helphost.exe      828    8      204800
          1413120      1/9/2004 2:52 PM      5.2.3790.0 (srv03_rtm.030324-2048) 106.00
KB (108,544 bytes)      5/30/2003 3:10 PM
helpctr.exe c:\windows\pchealth\helpctr\binaries\helpctr.exe 1772    8      204800
          1413120      1/9/2004 2:52 PM      5.2.3790.0 (srv03_rtm.030324-2048) 764.00
KB (782,336 bytes)      5/30/2003 3:10 PM
wmiprvse.exe Not Available 720    8      Not Available Not Available 1/9/2004 2:52
PM Not Available Not Available Not Available
  
```

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer	Path
winlogon	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\winlogon.exe
ntdll	5.2.3790.0 (srv03_rtm.030324-2048)	722.50 KB (739,840 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\ntdll.dll
kernel32	5.2.3790.0 (srv03_rtm.030324-2048)	965.00 KB (988,160 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\kernel32.dll
msvcrt	7.0.3790.0 (srv03_rtm.030324-2048)	319.50 KB (327,168 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\msvcrt.dll
advapi32	5.2.3790.0 (srv03_rtm.030324-2048)	559.50 KB (572,928 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\advapi32.dll
rpcrt4	5.2.3790.0 (srv03_rtm.030324-2048)	643.50 KB (658,944 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\rpcrt4.dll
user32	5.2.3790.0 (srv03_rtm.030324-2048)	562.00 KB (575,488 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\user32.dll
gdi32	5.2.3790.0 (srv03_rtm.030324-2048)	263.00 KB (269,312 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\gdi32.dll
userenv	5.2.3790.0 (srv03_rtm.030324-2048)	732.50 KB (750,080 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\userenv.dll
nddeapi	5.2.3790.0 (srv03_rtm.030324-2048)	16.00 KB (16,384 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\nddeapi.dll
crypt32	5.131.3790.0 (srv03_rtm.030324-2048)	598.00 KB (612,352 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\crypt32.dll
msasn1	5.2.3790.0 (srv03_rtm.030324-2048)	58.00 KB (59,392 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\msasn1.dll
secur32	5.2.3790.0 (srv03_rtm.030324-2048)	63.00 KB (64,512 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\secur32.dll
winsta	5.2.3790.0 (srv03_rtm.030324-2048)	51.00 KB (52,224 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\winsta.dll
netapi32	5.2.3790.0 (srv03_rtm.030324-2048)	317.00 KB (324,608 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\netapi32.dll
profmap	5.2.3790.0 (srv03_rtm.030324-2048)	22.00 KB (22,528 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\profmap.dll

Appendix C – Tunable Parameters

regapi 5.2.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\regapi.dll

ws2_32 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\ws2_32.dll

ws2help 5.2.3790.0 (srv03_rtm.030324-2048) 19.50 KB (19,968 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ws2help.dll

psapi 5.2.3790.0 (srv03_rtm.030324-2048) 21.50 KB (22,016 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\psapi.dll

version 5.2.3790.0 (srv03_rtm.030324-2048) 17.00 KB (17,408 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\version.dll

setupapi 5.2.3790.0 (srv03_rtm.030324-2048) 1,014.50 KB (1,038,848 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\setupapi.dll

msgina 5.2.3790.0 (srv03_rtm.030324-2048) 1.14 MB (1,191,936 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msgina.dll

shsvcs 6.00.3790.0 (srv03_rtm.030324-2048) 121.50 KB (124,416 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\shsvcs.dll

shlwapi 6.00.3790.0 (srv03_rtm.030324-2048) 281.00 KB (287,744 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\shlwapi.dll

sfc 5.2.3790.0 (srv03_rtm.030324-2048) 4.50 KB (4,608 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\sfc.dll

sfc_os 5.2.3790.0 (srv03_rtm.030324-2048) 133.00 KB (136,192 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\sfc_os.dll

wintrust 5.131.3790.0 (srv03_rtm.030324-2048) 161.50 KB (165,376 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wintrust.dll

ole32 5.2.3790.0 (srv03_rtm.030324-2048) 1.13 MB (1,187,328 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ole32.dll

imagehlp 5.2.3790.0 (srv03_rtm.030324-2048) 142.50 KB (145,920 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\imagehlp.dll

comctl32 6.0 (srv03_rtm.030324-2048) 907.00 KB (928,768 bytes) 5/30/2003 9:53
AM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-
controls_6595b64144ccf1df_6.0.100.0_x-ww_8417450b\comctl32.dll

winscard 5.2.3790.0 (srv03_rtm.030324-2048) 98.50 KB (100,864 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\winscard.dll

wtsapi32 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wtsapi32.dll

sxs 5.2.3790.0 (srv03_rtm.030324-2048) 733.00 KB (750,592 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\sxs.dll

winmm 5.2.3790.0 (srv03_rtm.030324-2048) 166.00 KB (169,984 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\winmm.dll

wldap32 5.2.3790.0 (srv03_rtm.030324-2048) 158.00 KB (161,792 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\wldap32.dll

rsaenh 5.2.3790.0 (srv03_rtm.030324-2048) 176.83 KB (181,072 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\rsaenh.dll

cscdll 5.2.3790.0 (srv03_rtm.030324-2048) 99.00 KB (101,376 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\cscdll.dll

wlnotify 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wlnotify.dll

winspool 5.2.3790.0 (srv03_rtm.030324-2048) 140.00 KB (143,360 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\winspool.drv

mpr 5.2.3790.0 (srv03_rtm.030324-2048) 56.00 KB (57,344 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\mpr.dll

shell32 6.00.3790.0 (srv03_rtm.030324-2048) 7.79 MB (8,166,400 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\shell32.dll

Appendix C – Tunable Parameters

comctl32 5.82 (srv03_rtm.030324-2048) 561.00 KB (574,464 bytes) 5/30/2003 9:53 AM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-controls_6595b64144ccf1df_5.82.0.0_x-ww_8a69ba05\comctl32.dll

uxtheme 6.00.3790.0 (srv03_rtm.030324-2048) 196.00 KB (200,704 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\uxtheme.dll

samlib 5.2.3790.0 (srv03_rtm.030324-2048) 49.00 KB (50,176 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\samlib.dll

cscui 5.2.3790.0 (srv03_rtm.030324-2048) 305.00 KB (312,320 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\cscui.dll

ntmarta 5.2.3790.0 (srv03_rtm.030324-2048) 114.00 KB (116,736 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\ntmarta.dll

oleaut32 5.2.3790.0 486.00 KB (497,664 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\oleaut32.dll

clbcatq 2001.12.4720.0 (srv03_rtm.030324-2048) 481.00 KB (492,544 bytes) 5/30/2003 3:07 PM Microsoft Corporation c:\windows\system32\clbcatq.dll

comres 2001.12.4720.0 (srv03_rtm.030324-2048) 778.00 KB (796,672 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\comres.dll

wbemprox 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes) 5/30/2003 3:07 PM Microsoft Corporation c:\windows\system32\wbem\wbemprox.dll

wbemcomn 5.2.3790.0 (srv03_rtm.030324-2048) 211.50 KB (216,576 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\wbem\wbemcomn.dll

wbemsvc 5.2.3790.0 (srv03_rtm.030324-2048) 42.50 KB (43,520 bytes) 5/30/2003 3:07 PM Microsoft Corporation c:\windows\system32\wbem\wbemsvc.dll

fastprox 5.2.3790.0 (srv03_rtm.030324-2048) 443.00 KB (453,632 bytes) 5/30/2003 3:07 PM Microsoft Corporation c:\windows\system32\wbem\fastprox.dll

msvcp60 6.05.2144.0 388.00 KB (397,312 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\msvcp60.dll

ntdsapi 5.2.3790.0 (srv03_rtm.030324-2048) 76.00 KB (77,824 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\ntdsapi.dll

dnsapi 5.2.3790.0 (srv03_rtm.030324-2048) 147.50 KB (151,040 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\dnsapi.dll

services 5.2.3790.0 (srv03_rtm.030324-2048) 102.00 KB (104,448 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\services.exe

scesrv 5.2.3790.0 (srv03_rtm.030324-2048) 316.50 KB (324,096 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\scesrv.dll

authz 5.2.3790.0 (srv03_rtm.030324-2048) 67.00 KB (68,608 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\authz.dll

umpnpgmr 5.2.3790.0 (srv03_rtm.030324-2048) 121.50 KB (124,416 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\umpnpgmr.dll

ncobjapi 5.2.3790.0 (srv03_rtm.030324-2048) 34.50 KB (35,328 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\ncobjapi.dll

eventlog 5.2.3790.0 (srv03_rtm.030324-2048) 60.50 KB (61,952 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\eventlog.dll

lsass 5.2.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\lsass.exe

lsasrv 5.2.3790.0 (srv03_rtm.030324-2048) 780.50 KB (799,232 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\lsasrv.dll

samsrv 5.2.3790.0 (srv03_rtm.030324-2048) 452.00 KB (462,848 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\samsrv.dll

cryptdll 5.2.3790.0 (srv03_rtm.030324-2048) 34.00 KB (34,816 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\cryptdll.dll

msprivs 5.2.3790.0 (srv03_rtm.030324-2048) 46.50 KB (47,616 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\msprivs.dll

Appendix C – Tunable Parameters

kerberos 5.2.3790.0 (srv03_rtm.030324-2048) 332.50 KB (340,480 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\kerberos.dll
msv1_05.2.3790.0 (srv03_rtm.030324-2048) 127.00 KB (130,048 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msv1_0.dll
netlogon 5.2.3790.0 (srv03_rtm.030324-2048) 409.00 KB (418,816 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\netlogon.dll
w32time 5.2.3790.0 (srv03_rtm.030324-2048) 216.00 KB (221,184 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\w32time.dll
iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\iphlpapi.dll
schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\schannel.dll
wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wdigest.dll
rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rassfm.dll
kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll
ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ntdsa.dll
ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\ntdsatq.dll
mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\mswsock.dll
esent 5.2.3790.0 (srv03_rtm.030324-2048) 1.01 MB (1,056,256 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\esent.dll
scecli 5.2.3790.0 (srv03_rtm.030324-2048) 179.50 KB (183,808 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\scecli.dll
wshtcpip 5.2.3790.0 (srv03_rtm.030324-2048) 18.00 KB (18,432 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wshtcpip.dll
dssenh 5.2.3790.0 (srv03_rtm.030324-2048) 131.33 KB (134,480 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\dssenh.dll
svchost 5.2.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\svchost.exe
rpcss 5.2.3790.0 (srv03_rtm.030324-2048) 276.50 KB (283,136 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\rpcss.dll
termsrv 5.2.3790.0 (srv03_rtm.030324-2048) 216.50 KB (221,696 bytes) 5/30/2003 3:07
PM Microsoft Corporation c:\windows\system32\termsrv.dll
icaapi 5.2.3790.0 (srv03_rtm.030324-2048) 10.50 KB (10,752 bytes) 5/30/2003 3:07 PM
Microsoft Corporation c:\windows\system32\icaapi.dll
mstlsapi 5.2.3790.0 (srv03_rtm.030324-2048) 104.50 KB (107,008 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\mstlsapi.dll
activeds 5.2.3790.0 (srv03_rtm.030324-2048) 189.00 KB (193,536 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\activeds.dll
adsldpc 5.2.3790.0 (srv03_rtm.030324-2048) 142.50 KB (145,920 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\adsldpc.dll
credui 5.2.3790.0 (srv03_rtm.030324-2048) 159.00 KB (162,816 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\credui.dll
atl 3.05.2283 83.00 KB (84,992 bytes) 3/29/2003 12:00 AM Microsoft Corporation
c:\windows\system32\atl.dll
rdpwsx 5.2.3790.0 (srv03_rtm.030324-2048) 80.13 KB (82,056 bytes) 5/30/2003 3:07 PM
Microsoft Corporation c:\windows\system32\rdpwsx.dll
wkssvc 5.2.3790.0 (srv03_rtm.030324-2048) 125.00 KB (128,000 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wkssvc.dll

Appendix C – Tunable Parameters

wiarpc 5.2.3790.0 (srv03_rtm.030324-2048) 30.00 KB (30,720 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wiarpc.dll

svrsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\svrsvc.dll

wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 5/30/2003 3:07
PM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll

vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\vssapi.dll

es 2001.12.4720.0 (srv03_rtm.030324-2048) 221.50 KB (226,816 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\es.dll

netman 5.2.3790.0 (srv03_rtm.030324-2048) 209.00 KB (214,016 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\netman.dll

mprapi 5.2.3790.0 (srv03_rtm.030324-2048) 81.00 KB (82,944 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\mprapi.dll

rtutils 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rtutils.dll

rasapi32 5.2.3790.0 (srv03_rtm.030324-2048) 227.50 KB (232,960 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\rasapi32.dll

rasman 5.2.3790.0 (srv03_rtm.030324-2048) 56.50 KB (57,856 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rasman.dll

tapi32 5.2.3790.0 (srv03_rtm.030324-2048) 175.00 KB (179,200 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\tapi32.dll

wzcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 272.50 KB (279,040 bytes) 3/25/2003 6:15
AM Microsoft Corporation c:\windows\system32\wzcsvc.dll

wmi 5.2.3790.0 (srv03_rtm.030324-2048) 6.50 KB (6,656 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wmi.dll

dhcpcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 101.50 KB (103,936 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\dhcpcsvc.dll

wzcsapi 5.2.3790.0 (srv03_rtm.030324-2048) 24.50 KB (25,088 bytes) 3/25/2003 6:15 AM
Microsoft Corporation c:\windows\system32\wzcsapi.dll

netshell 5.2.3790.0 (srv03_rtm.030324-2048) 1.67 MB (1,747,456 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\netshell.dll

clusapi 5.2.3790.0 (srv03_rtm.030324-2048) 56.00 KB (57,344 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\clusapi.dll

comsvcs 2001.12.4720.0 (srv03_rtm.030324-2048) 1.14 MB (1,199,616 bytes)
5/30/2003 3:07 PM Microsoft Corporation c:\windows\system32\comsvcs.dll

sens 5.2.3790.0 (srv03_rtm.030324-2048) 35.50 KB (36,352 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\sens.dll

hnetcfg 5.2.3790.0 (srv03_rtm.030324-2048) 243.50 KB (249,344 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\hnetcfg.dll

wininet 6.00.3790.0 (srv03_rtm.030324-2048) 609.00 KB (623,616 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wininet.dll

wbemcore 5.2.3790.0 (srv03_rtm.030324-2048) 457.00 KB (467,968 bytes)
5/30/2003 3:07 PM Microsoft Corporation
c:\windows\system32\wbem\wbemcore.dll

esscli 5.2.3790.0 (srv03_rtm.030324-2048) 235.50 KB (241,152 bytes) 5/30/2003 3:07
PM Microsoft Corporation c:\windows\system32\wbem\esscli.dll

wmiutils 5.2.3790.0 (srv03_rtm.030324-2048) 90.50 KB (92,672 bytes) 5/30/2003 3:07 PM
Microsoft Corporation c:\windows\system32\wbem\wmiutils.dll

repdrvfs 5.2.3790.0 (srv03_rtm.030324-2048) 165.00 KB (168,960 bytes)
5/30/2003 3:07 PM Microsoft Corporation c:\windows\system32\wbem\repdrvfs.dll

rasdlg 5.2.3790.0 (srv03_rtm.030324-2048) 642.00 KB (657,408 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\rasdlg.dll

Appendix C – Tunable Parameters

wmiprvsd	5.2.3790.0 (srv03_rtm.030324-2048)	405.50 KB (415,232 bytes)	
	5/30/2003 3:07 PM	Microsoft Corporation	c:\windows\system32\wbem\wmiprvsd.dll
wbemess	5.2.3790.0 (srv03_rtm.030324-2048)	256.50 KB (262,656 bytes)	
	5/30/2003 3:07 PM	Microsoft Corporation	c:\windows\system32\wbem\wbemess.dll
rasadhlp	5.2.3790.0 (srv03_rtm.030324-2048)	6.50 KB (6,656 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\rasadhlp.dll
ncprov	5.2.3790.0 (srv03_rtm.030324-2048)	43.00 KB (44,032 bytes)	5/30/2003 3:07 PM
		Microsoft Corporation	c:\windows\system32\wbem\ncprov.dll
dmserver	5.2.3790.0 (srv03_rtm.030324-2048)	24.00 KB (24,576 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\dmserver.dll
ntlsapi	5.2.3790.0 (srv03_rtm.030324-2048)	8.00 KB (8,192 bytes)	3/29/2003 12:00 AM
		Microsoft Corporation	c:\windows\system32\ntlsapi.dll
pchsvc	5.2.3790.0 (srv03_rtm.030324-2048)	31.50 KB (32,256 bytes)	5/30/2003 3:10 PM
		Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\pchsvc.dll
wbemcons	5.2.3790.0 (srv03_rtm.030324-2048)	69.00 KB (70,656 bytes)	5/30/2003 3:07 PM
		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)	
	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\explorer.exe
browseui	6.00.3790.0 (srv03_rtm.030324-2048)	1.01 MB (1,057,280 bytes)	
	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\browseui.dll
shdocvw	6.00.3790.0 (srv03_rtm.030324-2048)	1.33 MB (1,393,664 bytes)	
	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\shdocvw.dll
apphelp	5.2.3790.0 (srv03_rtm.030324-2048)	122.00 KB (124,928 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\apphelp.dll
themeui	6.00.3790.0 (srv03_rtm.030324-2048)	360.50 KB (369,152 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\themeui.dll
msimg32	5.2.3790.0 (srv03_rtm.030324-2048)	4.50 KB (4,608 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\msimg32.dll
linkinfo	5.2.3790.0 (srv03_rtm.030324-2048)	16.50 KB (16,896 bytes)	3/29/2003 12:00 AM
		Microsoft Corporation	c:\windows\system32\linkinfo.dll
ntshui	6.00.3790.0 (srv03_rtm.030324-2048)	136.00 KB (139,264 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\ntshui.dll
urlmon	6.00.3790.0 (srv03_rtm.030324-2048)	501.50 KB (513,536 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\urlmon.dll
webcheck	6.00.3790.0 (srv03_rtm.030324-2048)	261.50 KB (267,776 bytes)	
	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\webcheck.dll
wsock32	5.2.3790.0 (srv03_rtm.030324-2048)	22.00 KB (22,528 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\wsock32.dll
stobject	5.2.3790.0 (srv03_rtm.030324-2048)	117.50 KB (120,320 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\stobject.dll
batmeter	6.00.3790.0 (srv03_rtm.030324-2048)	28.50 KB (29,184 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\batmeter.dll
powrprof	6.00.3790.0 (srv03_rtm.030324-2048)	14.50 KB (14,848 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\powrprof.dll
printui	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\printui.dll
cfgmgr32	5.2.3790.0 (srv03_rtm.030324-2048)	17.50 KB (17,920 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\cfgmgr32.dll
drprov	5.2.3790.0 (srv03_rtm.030324-2048)	12.50 KB (12,800 bytes)	3/29/2003 12:00 AM
		Microsoft Corporation	c:\windows\system32\drprov.dll
ntlanman	5.2.3790.0 (srv03_rtm.030324-2048)	41.00 KB (41,984 bytes)	3/29/2003
	12:00 AM	Microsoft Corporation	c:\windows\system32\ntlanman.dll

Appendix C – Tunable Parameters

```

netui0 5.2.3790.0 (srv03_rtm.030324-2048) 75.50 KB (77,312 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\netui0.dll
netui1 5.2.3790.0 (srv03_rtm.030324-2048) 184.00 KB (188,416 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\netui1.dll
davclnt 5.2.3790.0 (srv03_rtm.030324-2048) 23.50 KB (24,064 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\davclnt.dll
browselc 6.00.3790.0 (srv03_rtm.030324-2048) 62.00 KB (63,488 bytes)3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\browselc.dll
shdoclc 6.00.3790.0 (srv03_rtm.030324-2048) 588.50 KB (602,624 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\shdoclc.dll
wzshlstb 3.0 (32-bit) 24.07 KB (24,644 bytes)4/19/2000 8:00 AM WinZip
Computing, Inc. c:\progra~1\winzip\wzshlstb.dll
zipfldr 6.00.3790.0 (srv03_rtm.030324-2048) 316.00 KB (323,584 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\zipfldr.dll
sendmail 6.00.3790.0 (srv03_rtm.030324-2048) 52.00 KB (53,248 bytes)3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\sendmail.dll
mydocs 6.00.3790.0 (srv03_rtm.030324-2048) 88.00 KB (90,112 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\mydocs.dll
actxprxy 6.00.3790.0 (srv03_rtm.030324-2048) 95.00 KB (97,280 bytes)3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\actxprxy.dll
sqlmangr 2000.080.0760.00 72.57 KB (74,308 bytes)11/7/2003 2:42 PM
Microsoft Corporation c:\program files\microsoft sql server\80\tools\bin\sqlmangr.exe
sqlunirl 2000.080.0728.00 176.56 KB (180,800 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\sqlunirl.dll
comdlg32 6.00.3790.0 (srv03_rtm.030324-2048) 261.00 KB (267,264 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\comdlg32.dll
w95scm 2000.080.0760.00 48.56 KB (49,728 bytes)11/7/2003 2:42 PM
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) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\odbc32.dll
sqlsvc 2000.080.0760.00 92.56 KB (94,784 bytes)11/7/2003 2:42 PM 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)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\odbcbc.dll
sqlresld 2000.080.0382.00 28.56 KB (29,248 bytes)11/7/2003 2:42 PM 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)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\odbcint.dll
resutils 5.2.3790.0 (srv03_rtm.030324-2048) 59.00 KB (60,416 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\resutils.dll
mfc42u 6.05.3014.0 960.00 KB (983,040 bytes) 3/29/2003 12:00 AM Microsoft
Corporation c:\windows\system32\mfc42u.dll
sqlsvc 2000.080.0194.00 24.00 KB (24,576 bytes)11/7/2003 2:42 PM 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)11/7/2003 2:42 PM
Microsoft Corporation c:\program files\microsoft sql
server\80\tools\bin\resources\1033\sqlmangr.rll
winvnc 3, 3, 3, 7 204.00 KB (208,896 bytes) 5/30/2003 4:28 PM AT&T Research
Labs Cambridge c:\program files\orl\vnc\winvnc.exe
vnchooks 3, 3, 3, 6 32.00 KB (32,768 bytes)5/30/2003 4:28 PM AT&T Research
Labs Cambridge c:\program files\orl\vnc\vnchooks.dll
omnithread_rt Not Available 44.00 KB (45,056 bytes)5/30/2003 4:28 PM Not Available
c:\windows\system32\omnithread_rt.dll
winrnr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\winrnr.dll

```

Appendix C – Tunable Parameters

mmc	5.2.3790.0 (srv03_rtm.030324-2048)	762.50 KB (780,800 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\mmc.exe	
oleacc	4.2.5406.0 (srv03_rtm.030324-2048)	171.00 KB (175,104 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\oleacc.dll	
mmcbase	5.2.3790.0 (srv03_rtm.030324-2048)	70.50 KB (72,192 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\mmcbase.dll	
mmcndmgr	5.2.3790.0 (srv03_rtm.030324-2048)	1.13 MB (1,182,720 bytes)	
3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\mmcndmgr.dll	
msxml3	8.40.9419.0	1.28 MB (1,337,344 bytes)	3/29/2003 12:00 AM
	Microsoft Corporation	c:\windows\system32\msxml3.dll	
sysmon	5.2.3790.0 (srv03_rtm.030324-2048)	241.50 KB (247,296 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\sysmon.ocx	
pdh	5.2.3790.0 (srv03_rtm.030324-2048)	274.50 KB (281,088 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\pdh.dll	
perfos	5.2.3790.0 (srv03_rtm.030324-2048)	24.50 KB (25,088 bytes)	3/29/2003 12:00 AM
	Microsoft Corporation	c:\windows\system32\perfos.dll	
perfdisk	5.2.3790.0 (srv03_rtm.030324-2048)	25.00 KB (25,600 bytes)	3/29/2003 12:00 AM
	Microsoft Corporation	c:\windows\system32\perfdisk.dll	
mlang	6.00.3790.0 (srv03_rtm.030324-2048)	570.00 KB (583,680 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\mlang.dll	
mscoree	1.1.4322.573	152.00 KB (155,648 bytes)	5/30/2003 3:07 PM
	Microsoft Corporation	c:\windows\system32\mscoree.dll	
PerfCounter	1.1.4322.573	88.00 KB (90,112 bytes)	5/30/2003 3:08 PM
	Microsoft Corporation	c:\windows\microsoft.net\framework\v1.1.4322\perfcounter.dll	
msvcr71	7.10.3052.4	340.00 KB (348,160 bytes)	5/30/2003 3:08 PM
	Microsoft Corporation	c:\windows\microsoft.net\framework\v1.1.4322\msvcr71.dll	
cmd	5.2.3790.0 (srv03_rtm.030324-2048)	374.00 KB (382,976 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\cmd.exe	
sqlservr	2000.080.0857.00	7.18 MB (7,532,584 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\sqlservr.exe	
opends60	2000.080.0194.00	24.06 KB (24,639 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\opends60.dll	
ums	2000.080.0760.00	52.55 KB (53,808 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ums.dll	
sqlsort	2000.080.0760.00	576.56 KB (590,396 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\sqlsort.dll	
msvcirt	7.0.3790.0 (srv03_rtm.030324-2048)	50.00 KB (51,200 bytes)	3/29/2003 12:00 AM
	Microsoft Corporation	c:\windows\system32\msvcirt.dll	
sqllevn70	2000.080.0760.00	28.00 KB (28,672 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\resources\1033\sqllevn70.rll	
xolehlp	2001.12.4720.0 (srv03_rtm.030324-2048)	8.50 KB (8,704 bytes)	5/30/2003 3:07 PM
	Microsoft Corporation	c:\windows\system32\xolehlp.dll	
msdtcprx	2001.12.4720.0 (srv03_rtm.030324-2048)	427.50 KB (437,760 bytes)	5/30/2003 3:07 PM
	Microsoft Corporation	c:\windows\system32\msdtcprx.dll	
mtxclu	2001.12.4720.0 (srv03_rtm.030324-2048)	74.50 KB (76,288 bytes)	3/29/2003
12:00 AM	Microsoft Corporation	c:\windows\system32\mtxclu.dll	
ssnetlib	2000.080.0851.00	80.07 KB (81,989 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ssnetlib.dll	
security	5.2.3790.0 (srv03_rtm.030324-2048)	5.50 KB (5,632 bytes)	3/29/2003 12:00 AM
	Microsoft Corporation	c:\windows\system32\security.dll	
ssmslpcn	2000.080.0760.00	28.56 KB (29,244 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ssmslpcn.dll	
ssnmpn70	2000.080.0534.00	24.56 KB (25,148 bytes)	11/7/2003 2:42 PM
	Microsoft Corporation	c:\program files\microsoft sql server\mssql\binn\ssnmpn70.dll	

Appendix C – Tunable Parameters

sqloledb 2000.085.1022.00 (srv03_rtm.030324-2048) 536.00 KB (548,864 bytes)
5/30/2003 3:10 PM 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) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msdart.dll

msdatl3 2.80.1022.0 (srv03_rtm.030324-2048) 96.00 KB (98,304 bytes) 5/30/2003 3:10 PM
Microsoft Corporation c:\program files\common files\system\ole db\msdatl3.dll

oledb32 2.80.1022.0 (srv03_rtm.030324-2048) 500.00 KB (512,000 bytes) 5/30/2003 3:10
PM 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) 5/30/2003 3:10
PM Microsoft Corporation c:\program files\common files\system\ole db\oledb32r.dll

xpstar 2000.080.0760.00 280.56 KB (287,296 bytes) 11/7/2003 2:42 PM
Microsoft Corporation c:\program files\microsoft sql server\mssql\binn\xpstar.dll

sqlresld 2000.080.0382.00 28.56 KB (29,248 bytes) 11/7/2003 2:42 PM Microsoft
Corporation c:\program files\microsoft sql server\mssql\binn\sqlresld.dll

sqlsvc 2000.080.0760.00 92.56 KB (94,784 bytes) 11/7/2003 2:42 PM Microsoft
Corporation c:\program files\microsoft sql server\mssql\binn\sqlsvc.dll

w95scm 2000.080.0760.00 48.56 KB (49,728 bytes) 11/7/2003 2:42 PM
Microsoft Corporation c:\program files\microsoft sql server\mssql\binn\w95scm.dll

shfolder 6.00.3790.0 (srv03_rtm.030324-2048) 23.00 KB (23,552 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\shfolder.dll

sqlsvc 2000.080.0194.00 24.00 KB (24,576 bytes) 11/7/2003 2:42 PM Microsoft
Corporation c:\program files\microsoft sql server\mssql\binn\resources\1033\sqlsvc.rll

xpstar 2000.080.0760.00 36.00 KB (36,864 bytes) 11/7/2003 2:42 PM Microsoft
Corporation c:\program files\microsoft sql server\mssql\binn\resources\1033\xpstar.rll

isql 2000.080.0194.00 96.00 KB (98,304 bytes) 11/7/2003 2:42 PM Microsoft
Corporation c:\progra~1\microso~1\80\tools\binn\isql.exe

ntwdblib 2000.080.0194.00 268.06 KB (274,489 bytes) 6/1/2003 1:13 PM
Microsoft Corporation c:\windows\system32\ntwdblib.dll

dbnetlib 2000.085.1022 (srv03_rtm.030324-2048) 76.00 KB (77,824 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\dbnetlib.dll

tail 5.2 build 63 43.50 KB (44,544 bytes) 6/2/2003 11:09 AM Mortice Kern Systems
Inc. c:\mks\mksnt\tail.exe

helpctr 5.2.3790.0 (srv03_rtm.030324-2048) 764.00 KB (782,336 bytes) 5/30/2003 3:10
PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\helpctr.exe

hcappres 5.2.3790.0 (srv03_rtm.030324-2048) 6.50 KB (6,656 bytes) 5/30/2003 3:10
PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\hcappres.dll

itss 5.2.3790.0 (srv03_rtm.030324-2048) 119.50 KB (122,368 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\itss.dll

pchshell 5.2.3790.0 (srv03_rtm.030324-2048) 100.50 KB (102,912 bytes)
5/30/2003 3:10 PM Microsoft Corporation
c:\windows\pchealth\helpctr\binaries\pchshell.dll

mshtml 6.00.3790.0 (srv03_rtm.030324-2048) 2.78 MB (2,916,352 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\mshtml.dll

msimtf 5.2.3790.0 (srv03_rtm.030324-2048) 149.00 KB (152,576 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msimtf.dll

msctf 5.2.3790.0 (srv03_rtm.030324-2048) 287.00 KB (293,888 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msctf.dll

jscript 5.6.0.8515 436.00 KB (446,464 bytes) 3/29/2003 12:00 AM Microsoft
Corporation c:\windows\system32\jscript.dll

mshtmlled 6.00.3790.0 (srv03_rtm.030324-2048) 443.50 KB (454,144 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\mshtmlled.dll

msls31 3.10.349.0 147.00 KB (150,528 bytes) 3/29/2003 12:00 AM Microsoft
Corporation c:\windows\system32\msls31.dll

Appendix C – Tunable Parameters

imm32 5.2.3790.0 (srv03_rtm.030324-2048) 105.50 KB (108,032 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\imm32.dll

imgutil 5.2.3790.0 (srv03_rtm.030324-2048) 35.00 KB (35,840 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\imgutil.dll

wbemdisp 5.2.3790.0 (srv03_rtm.030324-2048) 165.50 KB (169,472 bytes) 5/30/2003 3:07 PM Microsoft Corporation c:\windows\system32\wbem\wbemdisp.dll

wshom 5.6.0.8515 92.00 KB (94,208 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\wshom.ocx

scrrun 5.6.0.8515 148.00 KB (151,552 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\scrrun.dll

mfc42 6.05.3014.0 960.00 KB (983,040 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\mfc42.dll

helpsvc 5.2.3790.0 (srv03_rtm.030324-2048) 720.00 KB (737,280 bytes) 5/30/2003 3:10 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\helpsvc.exe

helphost 5.2.3790.0 (srv03_rtm.030324-2048) 106.00 KB (108,544 bytes) 5/30/2003 3:10 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\helphost.exe

sensapi 5.2.3790.0 (srv03_rtm.030324-2048) 6.00 KB (6,144 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\sensapi.dll

vbscript 5.6.0.8515 404.00 KB (413,696 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\vbscript.dll

msinfo 5.2.3790.0 (srv03_rtm.030324-2048) 358.50 KB (367,104 bytes) 5/30/2003 3:10 PM Microsoft Corporation c:\windows\pchealth\helpctr\binaries\msinfo.dll

riched32 5.2.3790.0 (srv03_rtm.030324-2048) 3.50 KB (3,584 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\riched32.dll

riched20 5.31.23.1218 406.00 KB (415,744 bytes) 3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\riched20.dll

[Services]

Display Name	Name	State	Start Mode	Service Type	Path	Error Control	Start
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	Manual 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	Disabled	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\dlhhost.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

Appendix C – Tunable Parameters

Distributed File System	Dfs	Stopped	Manual Own Process		
	c:\windows\system32\dfsrv.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\dmadmin.exe /com		Normal LocalSystem	0	
Logical Disk Manager	dmserver	Running	Manual 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		
	c:\windows\system32\imapi.exe		Normal LocalSystem	0	
Intersite Messaging	IsmServ	Stopped	Disabled Own Process		
	c:\windows\system32\ismserv.exe		Normal LocalSystem	0	
Kerberos Key Distribution Center	kdc	Stopped	Disabled Share Process		
	c:\windows\system32\lsass.exe		Normal LocalSystem	0	
Server	lanmanserver	Running	Auto Share Process		
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
Workstation	lanmanworkstation	Running	Auto Share Process		
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
License Logging	LicenseService	Stopped	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	mnmsrvc	Stopped	Disabled Own		
	c:\windows\system32\mnmsrvc.exe		Normal LocalSystem	0	
Distributed Transaction Coordinator	MSDTC	Stopped	Manual Own Process		
	c:\windows\system32\msdtc.exe		Normal NT AUTHORITY\NetworkService	0	
Windows Installer	MSIServer	Stopped	Manual Share Process		
	c:\windows\system32\msiexec.exe /v		Normal LocalSystem	0	
MSSQLSERVER	MSSQLSERVER	Stopped	Manual Own Process		
	c:\progra~1\microso~1\mssql\binn\sqlservr.exe		Normal LocalSystem	0	
MSSQLServerADHelper	MSSQLServerADHelper	Stopped	Manual Own Process		
	c:\program files\microsoft sql server\80\tools\binn\sqladhlp.exe		Normal LocalSystem	0	
0					
Network DDE	NetDDE	Stopped	Disabled Share Process		
	c:\windows\system32\netdde.exe		Normal LocalSystem	0	

Appendix C – Tunable Parameters

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	NetmanRunning		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
Ignore		LocalSystem	0	c:\windows\system32\ntfrs.exe
NT LM Security Support Provider	NtLmSsp	Stopped	Manual	Share Process
c:\windows\system32\lsass.exe		Normal	LocalSystem	0
Removable Storage	NtmsSvc	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Plug and Play	PlugPlay	Running	Auto	Share Process
c:\windows\system32\services.exe		Normal	LocalSystem	0
IPSEC ServicesPolicyAgent		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
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Remote Access Connection Manager	RasMan	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Remote Desktop Help Session Manager	RDSessMgr	Stopped	Manual	Own Process
c:\windows\system32\sessmgr.exe		Normal	LocalSystem	0
Routing and Remote Access	RemoteAccess	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Remote Registry	RemoteRegistry	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
c:\windows\system32\locator.exe		Normal	NT AUTHORITY\NetworkService	0
Remote Procedure Call (RPC)	RpcSs	Running	Auto	Share Process
c:\windows\system32\svchost -k rpcss		Normal	LocalSystem	0
Resultant Set of Policy Provider	RSOPProv	Stopped	Manual	Share Process
c:\windows\system32\rsopprov.exe		Normal	LocalSystem	0
Special Administration Console Helper	sacsvr	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Security Accounts Manager	SamSs	Stopped	Manual	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	seclogon	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Ignore	LocalSystem	0
System Event Notification	SENS	Running	Manual	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	c:\windows\system32\svchost.exe -k netsvcs	
Normal		LocalSystem	0	
Shell Hardware Detection	ShellHWDetection	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Ignore	LocalSystem	0

Appendix C – Tunable Parameters

Print Spooler	Spooler Stopped	Manual Own Process		
	c:\windows\system32\spoolsv.exe	Normal LocalSystem	0	
SQLSERVERAGENT	SQLSERVERAGENT	Stopped	Manual Own Process	
	c:\progra~1\microso~1\mssql\binn\sqlagent.exe	Normal LocalSystem	0	
Windows Image Acquisition (WIA)	stisvc	Stopped	Disabled	Share Process
	c:\windows\system32\svchost.exe -k imgsvc	Normal NT AUTHORITY\LocalService	0	
Microsoft Software Shadow Copy Provider	swprv	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 termsvcs	Normal LocalSystem	0	
Themes	Themes	Stopped	Disabled	Share Process
	c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem	0	
Telnet	TntSvr	Stopped	Disabled	Own Process
	c:\windows\system32\tlntsvr.exe	Normal NT AUTHORITY\LocalService	0	
Distributed Link Tracking Server	TrkSvr	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	wuauclt	Stopped	Manual Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem	0	
Wireless Configuration	WZC	Stopped	Manual Share Process	
	c:\windows\system32\svchost.exe -k netsvcs	Normal LocalSystem	0	

Appendix C – Tunable Parameters

[Program Groups]

Group Name	Name	User Name	Default User
Accessories	Default User:Accessories		Default User
Accessories\Accessibility	Default User:Accessories\Accessibility		Default User
Accessories\Entertainment	Default User:Accessories\Entertainment		Default User
Startup	Default User:Startup		Default User
Accessories	All Users:Accessories		All Users
Accessories\Accessibility	All Users:Accessories\Accessibility		All Users
Accessories\Communications	All Users:Accessories\Communications		All Users
Accessories\Entertainment	All Users:Accessories\Entertainment		All Users
Accessories\System Tools	All Users:Accessories\System Tools		All Users
Administrative Tools	All Users:Administrative Tools		All Users
Microsoft SQL Server	All Users:Microsoft SQL Server		All Users
MKS Toolkit	All Users:MKS Toolkit		All Users
Startup	All Users:Startup		All Users
VNC	All Users:VNC		All Users
VNC\Administrative Tools	All Users:VNC\Administrative Tools		All Users
WinZip	All Users:WinZip		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	PE2650\Administrator:Accessories		PE2650\Administrator
Accessories\Accessibility	PE2650\Administrator:Accessories\Accessibility		PE2650\Administrator
Accessories\Entertainment	PE2650\Administrator:Accessories\Entertainment		PE2650\Administrator
Administrative Tools	PE2650\Administrator:Administrative Tools		PE2650\Administrator
Startup	PE2650\Administrator:Startup		PE2650\Administrator

[Startup Programs]

Program	Command	User Name	Location
desktopdesktop.ini		NT AUTHORITY\SYSTEM	Startup
desktopdesktop.ini		PE2650\Administrator	Startup
Run WinVNC (App Mode)	run winvnc (app mode).lnk	PE2650\Administrator	Startup
desktopdesktop.ini	.DEFAULT		Startup
desktopdesktop.ini		All Users	Common Startup
Service Manager	c:\progra~1\microso~1\80\tools\binn\sqlmangr.exe /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

Appendix C – Tunable Parameters

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	
actxprxy.dll	6.0.3790.0	95 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
advpack.dll	6.0.3790.0	94 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
asctrls.ocx	6.0.3790.0	90 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
browseic.dll	6.0.3790.0	62 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
browseui.dll	6.0.3790.0	1,033 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
cdfview.dll	6.0.3790.0	144 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
comctl32.dll	5.82.3790.0	561 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
dxtrans.dll	6.3.3790.0	198 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
dxtmsft.dll	6.3.3790.0	344 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
iecont.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available	Not Available
iecontlc.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available	Not Available
iedkcs32.dll	16.0.3790.0	300 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
iepeers.dll	6.0.3790.0	230 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32
iesetup.dll	6.0.3790.0	59 KB	3/29/2003		Microsoft Corporation	C:\WINDOWS\system32

Appendix C – Tunable Parameters

ieunit.inf	Not Available	20 KB	3/29/2003	C:\WINDOWS\system32	Not Available
ieexplore.exe	6.0.3790.0 Microsoft Corporation	90 KB	3/29/2003	C:\Program Files\Internet Explorer	
imgutil.dll	5.2.3790.0 Microsoft Corporation	35 KB	3/29/2003	C:\WINDOWS\system32	
inetcpl.cpl	6.0.3790.0 Microsoft Corporation	303 KB	3/29/2003	C:\WINDOWS\system32	
inetcplc.dll	6.0.3790.0 Microsoft Corporation	109 KB	3/29/2003	C:\WINDOWS\system32	
inseng.dll	6.0.3790.0 Microsoft Corporation	72 KB	3/29/2003	C:\WINDOWS\system32	
mlang.dll	6.0.3790.0 Microsoft Corporation	570 KB	3/29/2003	C:\WINDOWS\system32	
msencode.dll	2002.10.4.0 Available	112 KB	3/29/2003	C:\WINDOWS\system32	Not Available
mshta.exe	6.0.3790.0 Microsoft Corporation	26 KB	3/29/2003	C:\WINDOWS\system32	
mshtml.dll	6.0.3790.0 Microsoft Corporation	2,848 KB	3/29/2003	C:\WINDOWS\system32	
mshtml.tlb	6.0.3790.0 Microsoft Corporation	1,319 KB	3/29/2003	C:\WINDOWS\system32	
mshtmlled.dll	6.0.3790.0 Microsoft Corporation	444 KB	3/29/2003	C:\WINDOWS\system32	
mshtmlr.dll	6.0.3790.0 Microsoft Corporation	55 KB	3/29/2003	C:\WINDOWS\system32	
msident.dll	6.0.3790.0 Microsoft Corporation	47 KB	3/29/2003	C:\WINDOWS\system32	
msidntld.dll	6.0.3790.0 Microsoft Corporation	15 KB	3/29/2003	C:\WINDOWS\system32	
msieftp.dll	6.0.3790.0 Microsoft Corporation	230 KB	3/29/2003	C:\WINDOWS\system32	
msrating.dll	6.0.3790.0 Microsoft Corporation	132 KB	3/29/2003	C:\WINDOWS\system32	
mstime.dll	6.0.3790.0 Microsoft Corporation	491 KB	3/29/2003	C:\WINDOWS\system32	
occache.dll	6.0.3790.0 Microsoft Corporation	89 KB	3/29/2003	C:\WINDOWS\system32	
proctexe.ocx	6.3.3790.0 Corporation	78 KB	3/29/2003	C:\WINDOWS\system32	Intel
sendmail.dll	6.0.3790.0 Microsoft Corporation	52 KB	3/29/2003	C:\WINDOWS\system32	
shdoclc.dll	6.0.3790.0 Microsoft Corporation	589 KB	3/29/2003	C:\WINDOWS\system32	
shdocvw.dll	6.0.3790.0 Microsoft Corporation	1,361 KB	3/29/2003	C:\WINDOWS\system32	
shfolder.dll	6.0.3790.0 Microsoft Corporation	23 KB	3/29/2003	C:\WINDOWS\system32	
shlwapi.dll	6.0.3790.0 Microsoft Corporation	281 KB	3/29/2003	C:\WINDOWS\system32	
tdc.ocx	1.3.0.3130 Corporation	58 KB	3/29/2003	C:\WINDOWS\system32	Microsoft
url.dll	6.0.3790.0 Corporation	36 KB	3/29/2003	C:\WINDOWS\system32	Microsoft

Appendix C – Tunable Parameters

urlmon.dll	6.0.3790.0	502 KB	3/29/2003	C:\WINDOWS\system32
	Microsoft Corporation			
webcheck.dll	6.0.3790.0	262 KB	3/29/2003	C:\WINDOWS\system32
	Microsoft Corporation			
wininet.dll	6.0.3790.0	609 KB	3/29/2003	C:\WINDOWS\system32
	Microsoft Corporation			

[Connectivity]

Item	Value
Connection Preference	Never dial

LAN Settings

AutoConfigProxy	Not Available
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:\WINDOWS\system32\config\systemprofile\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]

Appendix C – Tunable Parameters

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

Microsoft Windows 2003 Server System Info For PE1600SC

System Information report written at: 01/14/04 11:41:30

System Name: CLIENT77

[System Summary]

Item	Value
OS Name	Microsoft(R) Windows(R) Server 2003, Standard Edition
Version	5.2.3790 Build 3790
OS Manufacturer	Microsoft Corporation
System Name	CLIENT77
System Manufacturer	Dell Computer Corporation
System Model	PowerEdge 1600SC
System Type	X86-based PC
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2392 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2392 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2392 Mhz
Processor	x86 Family 15 Model 2 Stepping 7 GenuineIntel ~2392 Mhz
BIOS Version/Date	Dell Computer Corporation X22, 1/14/2003
SMBIOS Version	2.3
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "5.2.3790.0 (srv03_rtm.030324-2048)"
User Name	CLIENT77\Administrator
Time Zone	Central Standard Time
Total Physical Memory	1,024.00 MB
Available Physical Memory	818.53 MB
Total Virtual Memory	3.41 GB
Available Virtual Memory	3.10 GB
Page File Space	2.41 GB
Page File	C:\pagefile.sys

[Hardware Resources]

Appendix C – Tunable Parameters

[Conflicts/Sharing]

Resource	Device
I/O Port 0x00000000-0x000003AF	PCI bus
I/O Port 0x00000000-0x000003AF	Direct memory access controller
Memory Address 0xFD000000-0xFE1FFFFF	PCI bus
Memory Address 0xFD000000-0xFE1FFFFF	RAGE XL PCI Family (Microsoft Corporation)
Memory Address 0xA0000-0xBFFFF	PCI bus
Memory Address 0xA0000-0xBFFFF	RAGE XL PCI Family (Microsoft Corporation)
I/O Port 0x000003B0-0x000003DF	PCI bus
I/O Port 0x000003B0-0x000003DF	RAGE XL PCI Family (Microsoft Corporation)
Memory Address 0xFCB00000-0xFCDFFFFF	PCI bus
Memory Address 0xFCB00000-0xFCDFFFFF	Intel(R) PRO/100+ Server Adapter (PILA8470B)

[DMA]

Resource	Device	Status
Channel 4	Direct memory access controller	OK
Channel 2	Standard floppy disk controller	OK

[Forced Hardware]

Device PNP Device ID

[I/O]

Resource	Device	Status
0x00000000-0x000003AF	PCI bus	OK
0x00000000-0x000003AF	Direct memory access controller	OK
0x000003B0-0x000003DF	PCI bus	OK
0x000003B0-0x000003DF	RAGE XL PCI Family (Microsoft Corporation)	OK
0x000003E0-0x00000FFF	PCI bus	OK
0x0000E000-0x0000EFFF	PCI bus	OK
0x0000ECC0-0x0000ECFF	Intel(R) PRO/1000 MT Network Connection	OK
0x0000E800-0x0000E8FF	RAGE XL PCI Family (Microsoft Corporation)	OK
0x000003C0-0x000003DF	RAGE XL PCI Family (Microsoft Corporation)	OK
0x00000080-0x0000009F	Direct memory access controller	OK
0x000000C0-0x000000DF	Direct memory access controller	OK
0x000000F0-0x000000FF	Numeric data processor	OK
0x00000020-0x0000003F	Programmable interrupt controller	OK
0x000000A0-0x000000BF	Programmable interrupt controller	OK
0x000004D0-0x000004D1	Programmable interrupt controller	OK
0x00000061-0x00000061	System speaker	OK
0x00000040-0x0000005F	System timer	OK
0x000003F0-0x000003F5	Standard floppy disk controller	OK
0x000003F7-0x000003F7	Standard floppy disk controller	OK
0x00000060-0x00000060	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
0x00000064-0x00000064	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK

Appendix C – Tunable Parameters

0x000003F8-0x000003FF	Communications Port (COM1)	OK
0x00000378-0x0000037F	ECP Printer Port (LPT1)	OK
0x00000778-0x0000077F	ECP Printer Port (LPT1)	OK
0x00000070-0x0000007F	System CMOS/real time clock	OK
0x00000814-0x0000085B	System board	OK
0x00000820-0x0000083F	System board	OK
0x000008A0-0x000008AF	System board	OK
0x00000C00-0x00000CD7	System board	OK
0x00000F50-0x00000F58	System board	OK
0x000008B0-0x000008BF	Standard Dual Channel PCI IDE Controller	OK
0x000001F0-0x000001F7	Primary IDE Channel	OK
0x000003F6-0x000003F6	Primary IDE Channel	OK
0x00000170-0x00000177	Secondary IDE Channel	OK
0x00000376-0x00000376	Secondary IDE Channel	OK
0x00000A79-0x00000A79	ISAPNP Read Data Port	OK
0x00000279-0x00000279	ISAPNP Read Data Port	OK
0x00000274-0x00000277	ISAPNP Read Data Port	OK
0x0000D000-0x0000DFFF	PCI bus	OK
0x0000DC00-0x0000DCFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
0x0000C000-0x0000CFFF	PCI bus	OK
0x0000CCC0-0x0000CCFF	Intel(R) PRO/100+ Server Adapter (PILA8470B)	OK

[IRQs]

Resource	Device	Status
IRQ 9	Microsoft ACPI-Compliant System	OK
IRQ 16	Intel(R) PRO/1000 MT Network Connection	OK
IRQ 13	Numeric data processor	OK
IRQ 0	System timer	OK
IRQ 6	Standard floppy disk controller	OK
IRQ 1	Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	OK
IRQ 12	PS/2 Compatible Mouse	OK
IRQ 4	Communications Port (COM1)	OK
IRQ 8	System CMOS/real time clock	OK
IRQ 15	Secondary IDE Channel	OK
IRQ 10	ServerWorks (RCC) PCI to USB Open Host Controller	OK
IRQ 29	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
IRQ 24	Intel(R) PRO/100+ Server Adapter (PILA8470B)	OK

[Memory]

Resource	Device	Status
0xA0000-0xBFFFF	PCI bus	OK
0xA0000-0xBFFFF	RAGE XL PCI Family (Microsoft Corporation)	OK
0xD0000-0xE7FFF	PCI bus	OK
0xFD000000-0xFE1FFFFF	PCI bus	OK
0xFD000000-0xFE1FFFFF	RAGE XL PCI Family (Microsoft Corporation)	OK
0xFE100000-0xFE11FFFF	Intel(R) PRO/1000 MT Network Connection	OK
0xFE121000-0xFE121FFF	RAGE XL PCI Family (Microsoft Corporation)	OK
0xFE120000-0xFE120FFF	ServerWorks (RCC) PCI to USB Open Host Controller	OK
0x0000-0x9FFFF	System board	OK
0x100000-0x3FFFFFFF	System board	OK
0xF0000-0xFFFFF	System board	OK
0xFEC00000-0xFEC0FFFF	System board	OK
0xFEE00000-0xFEE0FFFF	System board	OK

Appendix C – Tunable Parameters

0xFFE00000-0xFFFFFFFF	System board	OK
0xFCE00000-0xFCFFFFFF	PCI bus	OK
0xFCF10000-0xFCF1FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
0xFCF00000-0xFCF0FFFF	LSI Logic PCI-X Ultra320 SCSI Host Adapter	OK
0xFCB00000-0xFCDFFFFFF	PCI bus	OK
0xFCB00000-0xFCDFFFFFF	Intel(R) PRO/100+ Server Adapter (PILA8470B)	OK
0xFCD00000-0xFCD00FFF	Intel(R) PRO/100+ Server Adapter (PILA8470B)	OK

[Components]

[Multimedia]

[Audio Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size	Creation Date
c:\windows\system32\sl_anet.acm	Sipro Lab Telecom Inc.	Sipro Lab Telecom Audio Codec	OK	C:\WINDOWS\system32\SL_ANET.ACM	3.02	84.00 KB (86,016 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msaud32.acm	Microsoft Corporation	Windows Media Audio Codec	OK	C:\WINDOWS\system32\MSAUD32.ACM	8.00.00.4487	288.00 KB (294,912 bytes)	3/29/2003 12:00 AM
c:\windows\system32\tsssoft32.acm	DSP GROUP, INC.		OK	C:\WINDOWS\system32\TSSOFT32.ACM	1.01	9.50 KB (9,728 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msadp32.acm	Microsoft Corporation		OK	C:\WINDOWS\system32\MSADP32.ACM	5.2.3790.0 (srv03_rtm.030324-2048)	14.50 KB (14,848 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msg711.acm	Microsoft Corporation		OK	C:\WINDOWS\system32\MSG711.ACM	5.2.3790.0 (srv03_rtm.030324-2048)	10.00 KB (10,240 bytes)	3/29/2003 12:00 AM
c:\windows\system32\msg723.acm	Microsoft Corporation		OK	C:\WINDOWS\system32\MSG723.ACM	4.4.4000	116.00 KB (118,784 bytes)	11/13/2003 1:32 PM
c:\windows\system32\msgsm32.acm	Microsoft Corporation		OK	C:\WINDOWS\system32\MSGSM32.ACM	5.2.3790.0 (srv03_rtm.030324-2048)	20.50 KB (20,992 bytes)	3/29/2003 12:00 AM
c:\windows\system32\l3codeca.acm	Fraunhofer Institut Integrierte Schaltungen IIS	Fraunhofer IIS MPEG Layer-3 Codec	OK	C:\WINDOWS\system32\L3CODECA.ACM	1, 9, 0, 0305	284.00 KB (290,816 bytes)	3/29/2003 12:00 AM
c:\windows\system32\imaadp32.acm	Microsoft Corporation		OK	C:\WINDOWS\system32\IMAADP32.ACM	5.2.3790.0 (srv03_rtm.030324-2048)	15.50 KB (15,872 bytes)	3/29/2003 12:00 AM

[Video Codecs]

CODEC	Manufacturer	Description	Status	File	Version	Size	Creation Date
c:\windows\system32\msyuv.dll	Microsoft Corporation		OK	C:\WINDOWS\system32\MSYUV.DLL	5.2.3790.0 (srv03_rtm.030324-2048)	16.50 KB (16,896 bytes)	3/24/2003 7:49 PM

Appendix C – Tunable Parameters

c:\windows\system32\msvidc32.dll Microsoft Corporation OK
C:\WINDOWS\system32\MSVIDC32.DLL 5.2.3790.0 (srv03_rtm.030324-2048)
26.50 KB (27,136 bytes) 3/29/2003 12:00 AM

c:\windows\system32\msh261.drv Microsoft Corporation OK
C:\WINDOWS\system32\MSH261.DRV 4.4.4000 180.00 KB (184,320 bytes)
11/13/2003 1:32 PM

c:\windows\system32\tsbyuv.dll Microsoft Corporation OK
C:\WINDOWS\system32\TSBYUV.DLL 5.2.3790.0 (srv03_rtm.030324-2048) 8.00 KB
(8,192 bytes) 3/24/2003 7:50 PM

c:\windows\system32\msrle32.dll Microsoft Corporation OK
C:\WINDOWS\system32\MSRLE32.DLL 5.2.3790.0 (srv03_rtm.030324-2048) 10.50
KB (10,752 bytes) 3/29/2003 12:00 AM

c:\windows\system32\iyuv_32.dll Microsoft Corporation OK
C:\WINDOWS\system32\IYUV_32.DLL 5.2.3790.0 (srv03_rtm.030324-2048) 45.00
KB (46,080 bytes) 3/24/2003 7:49 PM

c:\windows\system32\msh263.drv Microsoft Corporation OK
C:\WINDOWS\system32\MSH263.DRV 4.4.4000 284.00 KB (290,816 bytes)
3/24/2003 7:46 PM

[CD-ROM]

Item Value
Drive D:
Description CD-ROM Drive
Media Loaded No
Media Type CD-ROM
Name LITEON DVD-ROM LTD163
Manufacturer (Standard CD-ROM drives)
Status OK
Transfer Rate Not Available
SCSI Target ID 0
PNP Device ID IDE\CDROMLITEON_DVD-
ROM_LTD163_GDHB_5&1A6C219A&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), 3/29/2003 12:00 AM)

[Sound Device]

Item Value

[Display]

Item Value
Name RAGE XL PCI Family (Microsoft Corporation)
PNP Device ID PCI\VEN_1002&DEV_4752&SUBSYS_01351028&REV_27\3&13C0B0C5&0&70
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 65536
Resolution 1024 x 768 x 60 hertz
Bits/Pixel 16

Appendix C – Tunable Parameters

Memory Address 0xFD000000-0xFE1FFFFF
I/O Port 0x0000E800-0x0000E8FF
Memory Address 0xFE121000-0xFE121FFF
I/O Port 0x000003B0-0x000003DF
I/O Port 0x000003C0-0x000003DF
Memory Address 0xA0000-0xBFFFF
Driver c:\windows\system32\drivers\ati2mpad.sys (5.10.3663.6013, 335.38 KB (343,424 bytes), 11/13/2003 7:22 AM)

[Infrared]

Item Value

[Input]

[Keyboard]

Item Value
Description Standard 101/102-Key or Microsoft Natural PS/2 Keyboard
Name Enhanced (101- or 102-key)
Layout 00000409
PNP Device ID ACPI\PNP0303\4&25F73A82&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), 3/29/2003 12:00 AM)

[Pointing Device]

Item Value
Hardware Type PS/2 Compatible Mouse
Number of Buttons 3
Status OK
PNP Device ID ACPI\PNP0F13\4&25F73A82&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), 3/29/2003 12:00 AM)

[Modem]

Item Value

[Network]

[Adapter]

Appendix C – Tunable Parameters

Item Value
Name [00000001] Intel(R) PRO/1000 MT Network Connection
Adapter Type Ethernet 802.3
Product Type Intel(R) PRO/1000 MT Network Connection
Installed Yes
PNP Device ID PCI\VEN_8086&DEV_100E&SUBSYS_01351028&REV_02\3&13C0B0C5&0&10
Last Reset 1/14/2004 11:39 AM
Index 1
Service Name E1000
IP Address 192.1.100.77
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:C0:9F:1D:77:0D
Memory Address 0xFE100000-0xFE11FFFF
I/O Port 0x0000ECC0-0x0000ECFF
IRQ Channel IRQ 16
Driver c:\windows\system32\drivers\le1000325.sys (6.3.6.31 built by: WinDDK, 99.00 KB (101,376 bytes), 11/13/2003 7:22 AM)

Name [00000002] Intel(R) PRO/100+ Server Adapter (PILA8470B)
Adapter Type Ethernet 802.3
Product Type Intel(R) PRO/100+ Server Adapter (PILA8470B)
Installed Yes
PNP Device ID PCI\VEN_8086&DEV_1229&SUBSYS_100C8086&REV_08\3&29E81982&0&20
Last Reset 1/14/2004 11:39 AM
Index 2
Service Name E100B
IP Address 192.1.1.77
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:D0:B7:9E:A8:B9
Memory Address 0xFCD00000-0xFCD00FFF
I/O Port 0x0000CCC0-0x0000CCFF
Memory Address 0xFCB00000-0xFCDFFFFF
IRQ Channel IRQ 24
Driver c:\windows\system32\drivers\le100b325.sys (6.6.8.1 built by: WinDDK, 138.50 KB (141,824 bytes), 11/13/2003 7:22 AM)

Name [00000003] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type Not Available
Product Type Intel(R) PRO/100+ Dual Port Server Adapter
Installed Yes
PNP Device ID Not Available
Last Reset 1/14/2004 11:39 AM
Index 3
Service Name E100B
IP Address Not Available

Appendix C – Tunable Parameters

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 [00000004] Intel(R) PRO/100+ Dual Port Server Adapter
Adapter Type Not Available
Product Type Intel(R) PRO/100+ Dual Port Server Adapter
Installed Yes
PNP Device ID Not Available
Last Reset 1/14/2004 11:39 AM
Index 4
Service Name E100B
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 [00000005] RAS Async Adapter
Adapter Type Not Available
Product Type RAS Async Adapter
Installed Yes
PNP Device ID Not Available
Last Reset 1/14/2004 11:39 AM
Index 5
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 [00000006] WAN Miniport (L2TP)
Adapter Type Not Available
Product Type WAN Miniport (L2TP)
Installed Yes
PNP Device ID ROOTMS_L2TPMINIPORT\0000
Last Reset 1/14/2004 11:39 AM
Index 6
Service Name Rasl2tp
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available
DHCP Enabled No
DHCP Server Not Available

Appendix C – Tunable Parameters

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), 3/29/2003 12:00 AM)

Name [00000007] WAN Miniport (PPTP)
Adapter Type Wide Area Network (WAN)
Product Type WAN Miniport (PPTP)
Installed Yes
PNP Device ID ROOT\MS_PPTP\MINIPOINT\0000
Last Reset 1/14/2004 11:39 AM
Index 7
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\rasptp.sys (5.2.3790.0 (srv03_rtm.030324-2048), 70.50 KB (72,192 bytes), 3/29/2003 12:00 AM)

Name [00000008] WAN Miniport (PPPOE)
Adapter Type Wide Area Network (WAN)
Product Type WAN Miniport (PPPOE)
Installed Yes
PNP Device ID ROOT\MS_PPPOE\MINIPOINT\0000
Last Reset 1/14/2004 11:39 AM
Index 8
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\rasppoe.sys (5.2.3790.0 (srv03_rtm.030324-2048), 38.00 KB (38,912 bytes), 3/29/2003 12:00 AM)

Name [00000009] Direct Parallel
Adapter Type Not Available
Product Type Direct Parallel
Installed Yes
PNP Device ID ROOT\MS_PT\MINIPOINT\0000
Last Reset 1/14/2004 11:39 AM
Index 9
Service Name Raspti
IP Address Not Available
IP Subnet Not Available
Default IP Gateway Not Available

Appendix C – Tunable Parameters

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), 3/29/2003 12:00 AM)

Name [00000010] WAN Miniport (IP)
Adapter Type Not Available
Product Type WAN Miniport (IP)
Installed Yes
PNP Device ID ROOT\MS_NDISWANIP\0000
Last Reset 1/14/2004 11:39 AM
Index 10
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), 3/29/2003 12:00 AM)

[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

Appendix C – Tunable Parameters

Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting Yes

Name RSVP UDP Service Provider
Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 16 bytes
Maximum Message Size 63.93 KB (65,467 bytes)
Message Oriented Yes
Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption Yes
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting Yes

Name RSVP TCP Service Provider
Connectionless Service No
Guarantees Delivery Yes
Guarantees SequencingYes
Maximum Address Size 16 bytes
Maximum Message Size 0 bytes
Message Oriented No
Minimum Address Size 16 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption Yes
Supports Expedited Data Yes
Supports Graceful Closing Yes
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{833FA836-11FC-4387-9F56-9082062D338A}]
SEQPACKET 0
Connectionless Service No
Guarantees Delivery Yes
Guarantees SequencingYes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes

Appendix C – Tunable Parameters

Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{833FA836-11FC-4387-9F56-9082062D338A}]
DATAGRAM 0
Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{915E7CCF-F641-4E79-A07A-8707F1BF2257}]
SEQPACKET 1
Connectionless Service No
Guarantees Delivery Yes
Guarantees SequencingYes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{915E7CCF-F641-4E79-A07A-8707F1BF2257}]
DATAGRAM 1
Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 20 bytes

Appendix C – Tunable Parameters

Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{D92CDA13-25CE-4C24-8E01-DEE40289EF21}] SEQPACKET 2
Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{D92CDA13-25CE-4C24-8E01-DEE40289EF21}] DATAGRAM 2
Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{A92BF397-E0A5-448C-B3AE-8B34875D7FA4}] SEQPACKET 3
Connectionless Service No
Guarantees Delivery Yes

Appendix C – Tunable Parameters

Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{A92BF397-E0A5-448C-B3AE-8B34875D7FA4}] DATAGRAM 3

Connectionless Service Yes
Guarantees Delivery No
Guarantees Sequencing No
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{A1741045-84FE-47EA-904F-5DFCE3307C20}] SEQPACKET 4

Connectionless Service No
Guarantees Delivery Yes
Guarantees Sequencing Yes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{A1741045-84FE-47EA-904F-5DFCE3307C20}] DATAGRAM 4

Appendix C – Tunable Parameters

Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{F767E01D-2EAF-4E3A-AC93-B92C2D2CB4A4}] SEQPACKET 5

Connectionless Service No
Guarantees Delivery Yes
Guarantees SequencingYes
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting No
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Name MSAFD NetBIOS [\Device\NetBT_Tcpip_{F767E01D-2EAF-4E3A-AC93-B92C2D2CB4A4}] DATAGRAM 5

Connectionless Service Yes
Guarantees Delivery No
Guarantees SequencingNo
Maximum Address Size 20 bytes
Maximum Message Size 62.50 KB (64,000 bytes)
Message Oriented Yes
Minimum Address Size 20 bytes
Pseudo Stream Oriented No
Supports Broadcasting Yes
Supports Connect Data No
Supports Disconnect Data No
Supports Encryption No
Supports Expedited Data No
Supports Graceful Closing No
Supports Guaranteed Bandwidth No
Supports Multicasting No

Appendix C – Tunable Parameters

[WinSock]

Item Value
File c:\windows\system32\winsock.dll
Size 2.80 KB (2,864 bytes)
Version 3.10

File c:\windows\system32\wsock32.dll
Size 22.00 KB (22,528 bytes)
Version 5.2.3790.0 (srv03_rtm.030324-2048)

[Ports]

[Serial]

Item Value
Name Communications Port (COM1)
Status OK
PNP Device ID ACPI\PNP0501\1
Maximum Input Buffer Size 0
Maximum Output Buffer Size No
Settable Baud Rate Yes
Settable Data Bits Yes
Settable Flow Control Yes
Settable Parity Yes
Settable Parity Check Yes
Settable Stop Bits Yes
Settable RLSD Yes
Supports RLSD Yes
Supports 16 Bit Mode No
Supports Special Characters No
Baud Rate 9600
Bits/Byte 8
Stop Bits 1
Parity None
Busy No
Abort Read/Write on Error No
Binary Mode Enabled Yes
Continue XMit on XOff No
CTS Outflow Control No
Discard NULL Bytes No
DSR Outflow Control 0
DSR Sensitivity 0
DTR Flow Control Type Enable
EOF Character 0
Error Replace Character 0
Error Replacement Enabled No
Event Character 0
Parity Check Enabled No
RTS Flow Control Type Enable
XOff Character 19
XOffXMit Threshold 512
XOn Character 17

Appendix C – Tunable Parameters

XOnXMit Threshold 2048
XOnXOff InFlow Control0
XOnXOff OutFlow Control 0
I/O Port 0x000003F8-0x000003FF
IRQ Channel IRQ 4
Driver c:\windows\system32\drivers\serial.sys (5.2.3790.0 (srv03_rtm.030324-2048), 76.00 KB
(77,824 bytes), 3/29/2003 12:00 AM)

[Parallel]

Item Value
Name LPT1
PNP Device ID ACPI\PNP0401\4&25F73A82&0
I/O Port 0x00000378-0x0000037F
I/O Port 0x00000778-0x0000077F
Driver c:\windows\system32\drivers\parport.sys (5.2.3790.0 (srv03_rtm.030324-2048), 76.50 KB
(78,336 bytes), 3/24/2003 5:04 PM)

[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 7.81 GB (8,389,750,784 bytes)
Free Space 4.54 GB (4,874,027,008 bytes)
Volume Name
Volume Serial Number 98DEBC9B

Drive D:
Description CD-ROM Disc

Drive E:
Description Local Fixed Disk
Compressed No
File System NTFS
Size 26.10 GB (28,023,521,280 bytes)
Free Space 21.31 GB (22,880,321,536 bytes)
Volume Name Work
Volume Serial Number 5C407C96

[Disks]

Item Value
Description Disk drive
Manufacturer (Standard disk drives)
Model SEAGATE ST336752LW SCSI Disk Device

Appendix C – Tunable Parameters

Bytes/Sector 512
Media Loaded Yes
Media Type Fixed hard disk
Partitions 2
SCSI Bus 0
SCSI Logical Unit 0
SCSI Port 2
SCSI Target ID 0
Sectors/Track 63
Size 33.91 GB (36,413,314,560 bytes)
Total Cylinders 4,427
Total Sectors 71,119,755
Total Tracks 1,128,885
Tracks/Cylinder 255
Partition Disk #0, Partition #0
Partition Size 7.81 GB (8,389,753,344 bytes)
Partition Starting Offset 32,256 bytes
Partition Disk #0, Partition #1
Partition Size 26.10 GB (28,023,528,960 bytes)
Partition Starting Offset 8,389,785,600 bytes

[SCSI]

Item Value
Name LSI Logic PCI-X Ultra320 SCSI Host Adapter
Manufacturer LSI Logic Inc.
Status OK
PNP Device ID PCIIVEN_1000&DEV_0030&SUBSYS_01351028&REV_07\3&1070020&0&20
I/O Port 0x0000DC00-0x0000DCFF
Memory Address 0xFCF10000-0xFCF1FFFF
Memory Address 0xFCF00000-0xFCF0FFFF
IRQ Channel IRQ 29
Driver c:\windows\system32\drivers\symmpi.sys (1.08.18.00 (NT.021001-2000), 25.88 KB (26,496 bytes), 3/29/2003 12:00 AM)

[IDE]

Item Value
Name Standard Dual Channel PCI IDE Controller
Manufacturer (Standard IDE ATA/ATAPI controllers)
Status OK
PNP Device ID PCIIVEN_1166&DEV_0212&SUBSYS_41351028&REV_93\3&13C0B0C5&0&79
I/O Port 0x000008B0-0x000008BF
Driver c:\windows\system32\drivers\pciide.sys (5.2.3790.0 (srv03_rtm.030324-2048), 5.50 KB (5,632 bytes), 3/29/2003 12:00 AM)

Name Primary IDE Channel
Manufacturer (Standard IDE ATA/ATAPI controllers)
Status OK
PNP Device ID PCIIDE\IDECHANNEL\4&68D74DF&0&0
I/O Port 0x000001F0-0x000001F7
I/O Port 0x000003F6-0x000003F6
Driver c:\windows\system32\drivers\atapi.sys (5.2.3790.0 (srv03_rtm.030324-2048), 89.00 KB (91,136 bytes), 3/29/2003 12:00 AM)

Appendix C – Tunable Parameters

Name Secondary IDE Channel
 Manufacturer (Standard IDE ATA/ATAPI controllers)
 Status OK
 PNP Device ID PCI\IDE\IDECHANNEL\4&68D74DF&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), 3/29/2003 12:00 AM)

[Printing]

Name Driver Port Name Server Name

[Problem Devices]

Device PNP Device ID Error Code

[USB]

Device PNP Device ID
 ServerWorks (RCC) PCI to USB Open Host Controller
 PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_05\3&13C0B0C5&0&7A
 USB Root Hub USB\ROOT_HUB\4&1A0F8909&0

[Software Environment]

[System Drivers]

Name	Description	File	Type	Started	Start Mode	State	Status	Error Control
	Accept Pause	Accept Stop						
abiosdsk	Abiosdsk	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
acpi	Microsoft ACPI Driver	c:\windows\system32\drivers\acpi.sys	Kernel Driver	Yes	Running	OK	Normal	Yes
acpiec	ACPIEC	c:\windows\system32\drivers\acpiec.sys	Kernel Driver	No	Disabled	Stopped	OK	No
adpu160m	adpu160m	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
adpu320	adpu320	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
afcnt	afcnt	Not Available	Kernel Driver	No	Disabled	Stopped	Normal	OK
afd	AFD Networking Support Environment	c:\windows\system32\drivers\afd.sys	Kernel Driver	Yes	Running	OK	Normal	Yes
aha154x	Aha154x	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
aic78u2	aic78u2	Not Available	Kernel Driver	No	Disabled	Stopped	Normal	OK
aic78xx	aic78xx	Not Available	Kernel Driver	No	Disabled	Stopped	Normal	OK
aliide	Alilide	Not Available	Kernel Driver	No	Disabled	Stopped	Normal	OK

Appendix C – Tunable Parameters

asynmac	RAS Asynchronous Media Driver		c:\windows\system32\drivers\asynmac.sys	Kernel Driver	No	Manual		
	Stopped	OK	Normal	No	No			
atapi	Standard IDE/ESDI Hard Disk Controller		c:\windows\system32\drivers\atapi.sys	Kernel				
Driver	Yes	Boot	Running	OK	Normal	No	Yes	
atdisk	Atdisk	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
	Ignore	No	No					
ati2mpad	ati2mpad		c:\windows\system32\drivers\ati2mpad.sys	Kernel Driver				
	Yes	Manual	Running	OK	Ignore	No	Yes	
atmarpc	ATM ARP Client Protocol		c:\windows\system32\drivers\atmarpc.sys					
	Kernel Driver	No	Manual	Stopped	OK	Normal	No	No
audstub	Audio Stub Driver		c:\windows\system32\drivers\audstub.sys	Kernel Driver				
	Yes	Manual	Running	OK	Normal	No	Yes	
beep	Beep		c:\windows\system32\drivers\beep.sys	Kernel Driver	Yes	System		
	Running	OK	Normal	No	Yes			
cbidf2k	cbidf2k		c:\windows\system32\drivers\cbidf2k.sys	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No			
cd20xrnt	cd20xrnt		Not Available	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No			
cdfs	CdFs		c:\windows\system32\drivers\cdfs.sys	File System Driver		Yes		
	Disabled	Running	OK	Normal	No	Yes		
cdrom	CD-ROM Driver		c:\windows\system32\drivers\cdrom.sys	Kernel Driver	Yes	System		
	Running	OK	Normal	No	Yes			
changer	Changer		Not Available	Kernel Driver	No	System	Stopped	
	OK	Ignore	No	No				
clusdisk	Cluster Disk Driver		c:\windows\system32\drivers\clusdisk.sys	Kernel Driver				
	No	Disabled	Stopped	OK	Normal	No	No	
cmdide	CmdIde		Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No	No					
cpqarray	Cpqarray		Not Available	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No			
cpqarray2	cpqarray2		Not Available	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No			
cpqcissm	cpqcissm		Not Available	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No			
cpqfcalm	cpqfcalm		Not Available	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No			
crcdisk	CRC Disk Filter Driver		c:\windows\system32\drivers\crcdisk.sys	Kernel Driver	Yes			
	Boot	Running	OK	Normal	No	Yes		
dac960nt	dac960nt		Not Available	Kernel Driver	No	Disabled		
	Stopped	OK	Normal	No	No			
dellcerc	dellcerc		Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No	No					
dfsdriver	DfsDriver		c:\windows\system32\drivers\dfs.sys	File System Driver				
	Yes	Boot	Running	OK	Normal	No	Yes	
disk	Disk Driver		c:\windows\system32\drivers\disk.sys	Kernel Driver	Yes	Boot		
	Running	OK	Normal	No	Yes			
dmboot	dmboot		c:\windows\system32\drivers\dmboot.sys	Kernel Driver	No			
	Disabled	Stopped	OK	Normal	No	No		
dmio	Logical Disk Manager Driver		c:\windows\system32\drivers\dmio.sys	Kernel Driver				
	Yes	Boot	Running	OK	Normal	No	Yes	
dmload	dmload		c:\windows\system32\drivers\dmload.sys	Kernel Driver	Yes	Boot		
	Running	OK	Normal	No	Yes			
dpti2o	dpti2o		Not Available	Kernel Driver	No	Disabled	Stopped	OK
	Normal	No	No					

Appendix C – Tunable Parameters

e1000	Intel(R) PRO/1000 Device Driver	c:\windows\system32\drivers\e1000325.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes	
e100b	Intel(R) PRO Adapter Driver	c:\windows\system32\drivers\e100b325.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes	
fastfat	Fastfat	c:\windows\system32\drivers\fastfat.sys	File System Driver	Disabled	Stopped	OK	Normal	No	No		
fdc	Floppy Disk Controller Driver	c:\windows\system32\drivers\fdc.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes	
fips	Fips	c:\windows\system32\drivers\fips.sys	Kernel Driver	Running	OK	Normal	No	Yes		Yes	System
flpydisk	Floppy Disk Driver	c:\windows\system32\drivers\flpydisk.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes	
ftdisk	Volume Manager Driver	c:\windows\system32\drivers\ftdisk.sys	Kernel Driver	Boot	Running	OK	Normal	No	Yes		Yes
gpc	Generic Packet Classifier	c:\windows\system32\drivers\msgpc.sys	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes	
hpn	hpn	Not Available	Kernel Driver	Normal	No	No	Kernel Driver	No	Disabled	Stopped	OK
hpt3xx	hpt3xx	Not Available	Kernel Driver	Normal	No	No	Kernel Driver	No	Disabled	Stopped	OK
http	HTTP	c:\windows\system32\drivers\http.sys	Kernel Driver	Running	OK	Normal	No	Yes		Yes	Manual
i2omgmt	i2omgmt	Not Available	Kernel Driver	OK	Normal	No	Kernel Driver	No		System Stopped	
i2omp	i2omp	Not Available	Kernel Driver	Normal	No	No	Kernel Driver	No	Disabled	Stopped	OK
i8042prt	i8042 Keyboard and PS/2 Mouse Port Driver	c:\windows\system32\drivers\i8042prt.sys	Kernel Driver	Running	OK	Normal	No	Yes		Yes	System
iirsp	iirsp	Not Available	Kernel Driver	Normal	No	No	Kernel Driver	No	Disabled	Stopped	OK
imapi	CD-Burning Filter Driver	c:\windows\system32\drivers\imapi.sys	Kernel Driver	System Stopped	OK	Normal	No	No			No
intelide	IntelIde	Not Available	Kernel Driver	Normal	No	No	Kernel Driver	No	Disabled	Stopped	OK
ipfilterdriver	IP Traffic Filter Driver	c:\windows\system32\drivers\ipfltdrv.sys	Kernel Driver	No	Manual	Stopped	OK	Normal	No	No	
ipinip	IP in IP Tunnel Driver	c:\windows\system32\drivers\ipinip.sys	Kernel Driver	Manual	Stopped	OK	Normal	No	No		No
ipnat	IP Network Address Translator	c:\windows\system32\drivers\ipnat.sys	Kernel Driver	No	Manual	Stopped	OK	Normal	No	No	
ipsec	IPSEC driver	c:\windows\system32\drivers\ipsec.sys	Kernel Driver	Running	OK	Normal	No	Yes		Yes	System
ipsraidn	ipsraidn	Not Available	Kernel Driver	Normal	No	No	Kernel Driver	No	Disabled	Stopped	OK
irenum	IR Enumerator Service	c:\windows\system32\drivers\irenum.sys	Kernel Driver	Manual	Stopped	OK	Normal	No	No		No
isapnp	PnP ISA/EISA Bus Driver	c:\windows\system32\drivers\isapnp.sys	Kernel Driver	Yes	Boot	Running	OK	Critical	No	Yes	
kbdclass	Keyboard Class Driver	c:\windows\system32\drivers\kbdclass.sys	Kernel Driver	Yes	System	Running	OK	Normal	No	Yes	
ksecdd	KSecDD	c:\windows\system32\drivers\ksecdd.sys	Kernel Driver	Running	OK	Normal	No	Yes		Yes	Boot
lp6nds35	lp6nds35	Not Available	Kernel Driver	Stopped	OK	Normal	No	No		Disabled	

Appendix C – Tunable Parameters

mnmdd	mnmdd	c:\windows\system32\drivers\mnmdd.sys	Kernel Driver	Yes	System				
	Running	OK	Ignore	No	Yes				
modem	Modem	c:\windows\system32\drivers\modem.sys	Kernel Driver	No	Manual				
	Stopped	OK	Ignore	No	No				
mouclass	Mouse Class Driver	c:\windows\system32\drivers\mouclass.sys	Kernel						
Driver	Yes	System	Running	OK	Normal	No	Yes		
mountmgr	Mount Point Manager	c:\windows\system32\drivers\mountmgr.sys	Kernel						
Driver	Yes	Boot	Running	OK	Normal	No	Yes		
mraid3x	mraid3x	Not Available	Kernel Driver	No	Disabled				
	Stopped	OK	Normal	No	No				
mrxdav	WebDav Client Redirector	c:\windows\system32\drivers\mrxdav.sys	File						
System Driver	No	Manual	Stopped	OK	Normal	No	No		
mrxsmbr	MRXSMB	c:\windows\system32\drivers\mrxsmbr.sys	File System Driver						
	Yes	System	Running	OK	Normal	No	Yes		
msfs	Msfs	c:\windows\system32\drivers\msfs.sys	File System Driver	Yes	System				
	Running	OK	Normal	No	Yes				
mup	Mup	c:\windows\system32\drivers\mup.sys	File System Driver	Yes	Boot				
	Running	OK	Normal	No	Yes				
ndis	NDIS System Driver	c:\windows\system32\drivers\ndis.sys	Kernel Driver	Yes					
	Boot	Running	OK	Normal	No	Yes			
ndistapi	Remote Access NDIS TAPI Driver	c:\windows\system32\drivers\ndistapi.sys							
	Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes	
ndisuio	NDIS Usermode I/O Protocol	c:\windows\system32\drivers\ndisuio.sys	Kernel Driver						
	Yes	Manual	Running	OK	Normal	No	Yes		
ndiswan	Remote Access NDIS WAN Driver	c:\windows\system32\drivers\ndiswan.sys	Kernel Driver	Yes	Manual				
	Running	OK	Normal	No	Yes				
ndproxy	NDIS Proxy	c:\windows\system32\drivers\ndproxy.sys	Kernel Driver	Yes					
	Manual	Running	OK	Normal	No	Yes			
netbios	NetBIOS Interface	c:\windows\system32\drivers\netbios.sys	File System						
Driver	Yes	System	Running	OK	Normal	No	Yes		
netbt	NetBios over Tcpip	c:\windows\system32\drivers\netbt.sys	Kernel Driver	Yes					
	System	Running	OK	Normal	No	Yes			
nfrd960	nfrd960	Not Available	Kernel Driver	No	Disabled	Stopped	OK		
	Normal	No	No						
npfs	Npfs	c:\windows\system32\drivers\npfs.sys	File System Driver	Yes	System				
	Running	OK	Normal	No	Yes				
ntfs	Ntfs	c:\windows\system32\drivers\ntfs.sys	File System Driver	Yes					
	Disabled	Running	OK	Normal	No	Yes			
null	Null	c:\windows\system32\drivers\null.sys	Kernel Driver	Yes	System				
	Running	OK	Normal	No	Yes				
parport	Parallel port driver	c:\windows\system32\drivers\parport.sys	Kernel Driver	Yes					
	Manual	Running	OK	Normal	No	Yes			
partmgr	Partition Manager	c:\windows\system32\drivers\partmgr.sys	Kernel Driver						
	Yes	Boot	Running	OK	Normal	No	Yes		
parvdm	Parvdm	c:\windows\system32\drivers\parvdm.sys	Kernel Driver	Yes	Auto				
	Running	OK	Ignore	No	Yes				
pci	PCI Bus Driver	c:\windows\system32\drivers\pci.sys	Kernel Driver	Yes	Boot				
	Running	OK	Critical	No	Yes				
pciide	PCIIde	c:\windows\system32\drivers\pciide.sys	Kernel Driver	Yes	Boot				
	Running	OK	Normal	No	Yes				
pcmcia	Pcmcia	c:\windows\system32\drivers\pcmcia.sys	Kernel Driver	No	Disabled				
	Stopped	OK	Normal	No	No				
pdcomp	PDCOMP	Not Available	Kernel Driver	No	Manual	Stopped	OK		
	Ignore	No	No						

Appendix C – Tunable Parameters

pdframe	PDFRAME	Not Available	Kernel Driver	No	Manual	Stopped	
OK	Ignore	No	No				
pdreli	PDRELI	Not Available	Kernel Driver	No	Manual	Stopped	OK
Ignore	No	No					
pdframe	PDRFRAME	Not Available	Kernel Driver	No	Manual	Stopped	
OK	Ignore	No	No				
perc2	perc2	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
perc2hib	perc2hib	Not Available	Kernel Driver	No	Disabled		
Stopped	OK	Normal	No	No			
pptpminiport	WAN Miniport (PPTP)		c:\windows\system32\drivers\rasptp.sys				Kernel
Driver	Yes	Manual	Running	OK	Normal	No	Yes
processor	Processor Driver		c:\windows\system32\drivers\processr.sys				Kernel
Driver	Yes	Manual	Running	OK	Normal	No	Yes
ptilink	Direct	Parallel Link	Driver	c:\windows\system32\drivers\ptilink.sys			Kernel Driver
Yes	Manual	Running		OK	Normal	No	Yes
ql1080	ql1080	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
ql10wnt	QL10wnt	Not Available	Kernel Driver	No	Disabled	Stopped	
OK	Normal	No	No				
ql12160	ql12160	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
ql1240	ql1240	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
ql1280	ql1280	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
ql2100	ql2100	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
ql2200	ql2200	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
ql2300	ql2300	Not Available	Kernel Driver	No	Disabled	Stopped	OK
Normal	No	No					
rasacd	Remote Access Auto Connection Driver		c:\windows\system32\drivers\rasacd.sys				Kernel
Driver	Yes	System	Running	OK	Normal	No	Yes
rasl2tp	WAN Miniport (L2TP)		c:\windows\system32\drivers\rasl2tp.sys				Kernel Driver
Manual	Running		OK	Normal	No	Yes	Yes
rasppoe	Remote Access PPPOE Driver		c:\windows\system32\drivers\rasppoe.sys				
Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes
raspti	Direct	Parallel	c:\windows\system32\drivers\raspti.sys				Kernel Driver
Running	OK	Normal	No	Yes			Yes
Manual							Manual
rbss	Rdbss	c:\windows\system32\drivers\rdbss.sys		File System	Driver	Yes	System
Running	OK	Normal	No	Yes			
rdpcdd	RDPCDD	c:\windows\system32\drivers\rdpcdd.sys		Kernel Driver	Yes		System
Running	OK	Ignore	No	Yes			
rdpdr	Terminal Server Device Redirector Driver		c:\windows\system32\drivers\rdpdr.sys				
Kernel Driver	Yes	Manual	Running	OK	Normal	No	Yes
rdpwd	RDPWD	c:\windows\system32\drivers\rdpwd.sys		Kernel Driver	Yes		Manual
Running	OK	Ignore	No	Yes			
redbook	Digital CD Audio Playback Filter Driver		c:\windows\system32\drivers\redbook.sys				Kernel Driver
Running	OK	Normal	No	Yes	Yes		System
secdrv	Secdrv	c:\windows\system32\drivers\secdrv.sys		Kernel Driver	No		Manual
Stopped	OK	Normal	No	No			
serenum	Serenum Filter Driver		c:\windows\system32\drivers\serenum.sys				Kernel
Driver	Yes	Manual	Running	OK	Normal	No	Yes

Appendix C – Tunable Parameters

serial	Serial port driver	c:\windows\system32\drivers\serial.sys	Kernel Driver	Yes				
	System Running	OK	Ignore	No	Yes			
sfloppy	Sfloppy	c:\windows\system32\drivers\sfloppy.sys	Kernel Driver	No	System			
	Stopped	OK	Ignore	No	No			
simbad	Simbad	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
	Normal	No	No					
sparrow	Sparrow	Not Available	Kernel Driver	No	Disabled	Stopped		
	OK	Normal	No	No				
srv	Srv	c:\windows\system32\drivers\srv.sys	File System Driver	Yes	Manual			
	Running	OK	Normal	No	Yes			
swenum	Software Bus Driver	c:\windows\system32\drivers\swenum.sys			Kernel			
Driver	Yes	Manual	Running	OK	Normal	No	Yes	
symc810	symc810	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
symc8xx	symc8xx	Not Available	Kernel Driver	No	Disabled			
	Stopped	OK	Normal	No	No			
symmpi	symmpi	c:\windows\system32\drivers\symmpi.sys	Kernel Driver	Yes	Boot			
	Running	OK	Normal	No	Yes			
sym_hi	sym_hi	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
	Normal	No	No					
sym_u3	sym_u3	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
	Normal	No	No					
tcpip	TCP/IP Protocol Driver	c:\windows\system32\drivers\tcpip.sys	Kernel Driver	Yes				
	System Running	OK	Normal	No	Yes			
tdpipe	TDPIPE	c:\windows\system32\drivers\tdpipe.sys	Kernel Driver	No	Manual			
	Stopped	OK	Ignore	No	No			
tdtcp	TDTCP	c:\windows\system32\drivers\tdtcp.sys	Kernel Driver	Yes	Manual			
	Running	OK	Ignore	No	Yes			
termdd	Terminal Device Driver	c:\windows\system32\drivers\termdd.sys	Kernel Driver	Yes				
	System Running	OK	Normal	No	Yes			
toside	Toside	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
	Normal	No	No					
udfs	Udfs	c:\windows\system32\drivers\udfs.sys	File System Driver	No				
	Disabled	Stopped	OK	Normal	No	No		
ultra	ultra	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
	Normal	No	No					
update	Microcode Update Driver	c:\windows\system32\drivers\update.sys	Kernel Driver					
	Yes	Manual	Running	OK	Normal	No	Yes	
usbhub	USB2 Enabled Hub	c:\windows\system32\drivers\usbhub.sys			Kernel Driver			
	Yes	Manual	Running	OK	Normal	No	Yes	
usbohci	Microsoft USB Open Host Controller Miniport Driver	c:\windows\system32\drivers\usbohci.sys	Kernel Driver	Yes	Manual			
	Running	OK	Normal	No	Yes			
vgasave	VGA Display Controller.	c:\windows\system32\drivers\vga.sys	Kernel Driver					
	Yes	System Running	OK	Ignore	No	Yes		
viaide	Vialde	Not Available	Kernel Driver	No	Disabled	Stopped	OK	
	Normal	No	No					
volsnap	Storage volumes	c:\windows\system32\drivers\volsnap.sys	Kernel Driver					
	Yes	Boot	Running	OK	Normal	No	Yes	
wanarp	Remote Access IP ARP Driver	c:\windows\system32\drivers\wanarp.sys			Kernel			
Driver	Yes	Manual	Running	OK	Normal	No	Yes	
wdica	WDICA	Not Available	Kernel Driver	No	Manual	Stopped	OK	Ignore
	No	No						
wlbs	Network Load Balancing	c:\windows\system32\drivers\wlbs.sys	Kernel Driver					
	No	Manual	Stopped	OK	Normal	No	No	

Appendix C – Tunable Parameters

[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	Not Available	Not Available	Not Available
		Not Available	HTREE\ROOT\0			
ACPI Multiprocessor (Standard computers)	Yes	COMPUTER	5.2.3790.0	10/1/2002	(Standard computers)	hal.inf
Microsoft ACPI-Compliant System	Yes	SYSTEM	5.2.3790.0	10/1/2002		acpi.inf
Microsoft Processor (Standard processor types)	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf
Processor	Yes	PROCESSOR	5.2.3790.0	10/1/2002	(Standard processor types)	cpu.inf
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf
ServerWorks Grand Champion	Yes	CMIC_SL - NorthBridge	5.2.3790.0	10/1/2002	Super Lite	machine.inf
ServerWorks Grand Champion	Yes	CMIC_SL - NorthBridge	5.2.3790.0	10/1/2002	Super Lite	machine.inf
Intel(R) PRO/1000 MT Network Connection	Yes	NET	6.3.6.31	10/1/2002		nete1000.inf
RAGE XL PCI Family (Microsoft Corporation)	Yes	DISPLAY	5.10.2600.6014	8/8/2001		atiixpad.inf
Default Monitor (Standard monitor types)	Yes	MONITOR	5.1.2001.0	6/6/2001	(Standard monitor types)	monitor.inf
ServerWorks Champion CSB5 - SouthBridge	Yes	SYSTEM	5.2.3790.0	10/1/2002		machine.inf
Direct memory access controller	Yes	SYSTEM	5.2.3790.0	10/1/2002		machine.inf
Numeric data processor (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf
Programmable interrupt controller (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf
System speaker (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf
System timer (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)	machine.inf
Standard floppy disk controller (Standard floppy disk controllers)	Yes	FDC	5.2.3790.0	10/1/2002	(Standard floppy disk controllers)	fdc.inf

Appendix C – Tunable Parameters

Floppy disk drive floppy disk drives)	Yes	FLOPPYDISK	5.2.3790.0	10/1/2002	(Standard
		flpydisk.inf	Not Available		
		FDC\GENERIC_FLOPPY_DRIVE\5&1AE2F47D&0&0			
Standard 101/102-Key or Microsoft Natural PS/2 Keyboard	Yes	KEYBOARD			
		5.2.3790.0	10/1/2002	(Standard keyboards)	keyboard.inf
		ACPI\PNP0303\4&25F73A82&0			
PS/2 Compatible Mouse	Yes	MOUSE	5.2.3790.0	10/1/2002	Microsoft
		msmouse.inf	Not Available		ACPI\PNP0F13\4&25F73A82&0
Communications Port	Yes	PORTS	5.2.3790.0	10/1/2002	(Standard port types)
		msports.inf	Not Available		ACPI\PNP0501\1
ECP Printer Port	Yes	PORTS	5.2.3790.0	10/1/2002	(Standard port types)
		msports.inf	Not Available		ACPI\PNP0401\4&25F73A82&0
Printer Port Logical Interface	Yes	SYSTEM	5.2.3790.0	10/1/2002	
(Standard system devices)		machine.inf	Not Available		
		LPTENUMMICROSOFTRAWPORT\5&39F3CAEA&0&LPT1			
System CMOS/real time clock	Yes	SYSTEM	5.2.3790.0	10/1/2002	
(Standard system devices)		machine.inf	Not Available		
		ACPI\PNP0B00\4&25F73A82&0			
System board	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system
devices)		machine.inf	Not Available		ACPI\PNP0C01\2
Standard Dual Channel PCI IDE Controller	Yes	HDC	5.2.3790.0	10/1/2002	
(Standard IDE ATA/ATAPI controllers)		mshdc.inf	Not Available		
		PCI\VEN_1166&DEV_0212&SUBSYS_41351028&REV_93\3&13C0B0C5&0&79			
Primary IDE Channel	Yes	HDC	5.2.3790.0	10/1/2002	(Standard IDE
ATA/ATAPI controllers)		mshdc.inf	Not Available		
		PCI\IDE\IDECHANNEL\4&68D74DF&0&0			
Secondary IDE Channel	Yes	HDC	5.2.3790.0	10/1/2002	(Standard IDE
ATA/ATAPI controllers)		mshdc.inf	Not Available		
		PCI\IDE\IDECHANNEL\4&68D74DF&0&1			
CD-ROM Drive	Yes	CDROM	5.2.3790.0	10/1/2002	(Standard CD-ROM
drives)		cdrom.inf	Not Available		IDE\CDROMLITEON_DVD-
		ROM_LTD163_____GDHB_____\5&1A6C219A&0&0.0.0			
ServerWorks (RCC) PCI to USB Open Host Controller	Yes	USB	5.2.3790.0		
		10/1/2002	ServerWorks (RCC)	usbport.inf	Not Available
		PCI\VEN_1166&DEV_0220&SUBSYS_02201166&REV_05\3&13C0B0C5&0&7A			
USB Root Hub	Yes	USB	5.2.3790.0	10/1/2002	(Standard USB Host Controller)
		usbport.inf	Not Available		USB\ROOT_HUB\4&1A0F8909&0
Serverworks Champion CSB5 - SouthBridge 5 LPC	Yes	SYSTEM	5.2.3790.0		
		10/1/2002	ServerWorks (RCC)	machine.inf	Not Available
		PCI\VEN_1166&DEV_0225&SUBSYS_00000000&REV_00\3&13C0B0C5&0&7B			
ISAPNP Read Data Port	Yes	SYSTEM	5.2.3790.0	10/1/2002	
(Standard system devices)		machine.inf	Not Available		
		ISAPNP\READDATAPORT\0			
ServerWorks Grand Champion CIOB_X2 - I/O Bridge 133 Mhz	Yes	SYSTEM			
		5.2.3790.0	10/1/2002	ServerWorks (RCC)	machine.inf
		PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV_03\3&13C0B0C5&0&80			
ServerWorks Grand Champion CIOB_X2 - I/O Bridge 133 Mhz	Yes	SYSTEM			
		5.2.3790.0	10/1/2002	ServerWorks (RCC)	machine.inf
		PCI\VEN_1166&DEV_0101&SUBSYS_00000000&REV_03\3&13C0B0C5&0&82			
System board	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system
devices)		machine.inf	Not Available		ACPI\PNP0C01\1
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
		machine.inf	Not Available		ACPI\PNP0A03\2

Appendix C – Tunable Parameters

LSI Logic PCI-X Ultra320 SCSI Host Adapter	Yes	SCSIADAPTER	5.2.3790.0		
10/1/2002	LSI Logic Inc.	pnpscsi.inf	Not Available		
	PCI\VEN_1000&DEV_0030&SUBSYS_01351028&REV_07\3&1070020&0&20				
Disk drive	Yes	DISKDRIVE	5.2.3790.0	10/1/2002	(Standard disk drives)
	disk.inf	Not Available			
	SCSI\DISK&VEN_SEAGATE&PROD_ST336752LW&REV_2212\4&1E63B2AC&0&000				
PCI bus	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
	machine.inf	Not Available	ACPI\PNP0A03\3		
Intel(R) PRO/100+ Server Adapter (PILA8470B)	Yes	NET	6.6.8.1	10/1/2002	Intel
	net557.inf	Not Available			
	PCI\VEN_8086&DEV_1229&SUBSYS_100C8086&REV_08\3&29E81982&0&20				
ACPI Fixed Feature Button	Yes	SYSTEM	5.2.3790.0	10/1/2002	
	(Standard system devices)	machine.inf	Not Available		
	ACPI\FIXEDBUTTON\2&DABA3FF&0				
Logical Disk Manager	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
	machine.inf	Not Available	ROOT\DMIO\0000		
Volume Manager	Yes	SYSTEM	5.2.3790.0	10/1/2002	(Standard system devices)
	machine.inf	Not Available	ROOT\FTDISK\0000		
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
	STORAGE\VOLUME\1&30A96598&0&SIGNATUREA7EFA7EFOFFSET7E00LENGTH1F4117A00				
Generic volume	Yes	VOLUME	5.2.3790.0	10/1/2002	Microsoft
	volume.inf	Not Available			
	STORAGE\VOLUME\1&30A96598&0&SIGNATUREA7EFA7EFOFFSET1F411F800LENGT				
	H686551E00				
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
	Not Available	Not Available	ROOT\LEGACY_BEEP\0000		
CRC Disk Filter Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_CRCDISK\0000		
dmbboot	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_DMBOOT\0000		
dmload	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_DMLOAD\0000		
Fips	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_FIPS\0000		
Generic Packet Classifier	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	Not Available	ROOT\LEGACY_GPC\0000	
HTTP	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_HTTP\0000		
IPSEC driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_IPSEC\0000		
ksecdd	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_KSECDD\0000		
mnmdd	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_MNMDD\0000		
mountmgr	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	ROOT\LEGACY_MOUNTMGR\0000		
NDIS System Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
	Not Available	Not Available	Not Available	ROOT\LEGACY_NDIS\0000	

Appendix C – Tunable Parameters

Remote Access NDIS TAPI Driver	Not Available	LEGACYDRIVER	Not Available		
Not Available	Not Available	Not Available	Not Available		
ROOT\LEGACY_NDISTAPI\0000					
NDIS Usermode I/O Protocol	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Available	Not Available	Not Available	Not Available		
ROOT\LEGACY_NDISUIO\0000					
NDProxy	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Available	Not Available	Not Available	ROOT\LEGACY_NDPROXY\0000		
NetBios over Tcpip	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_NETBT\0000		
Null	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_NULL\0000		
Partition Manager	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_PARTMGR\0000		
Parvdm	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_PARVDM\0000		
Remote Access Auto Connection Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	Not Available		
ROOT\LEGACY_RASACD\0000					
RDPcdd	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Available	Not Available	Not Available	ROOT\LEGACY_RDPcdd\0000		
RDPWD	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Available	Not Available	Not Available	ROOT\LEGACY_RDPWD\0000		
TCP/IP Protocol Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_TCPIP\0000		
TDTCP	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_TDTCP\0000		
VGA Display Controller.	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_VGASAVE\0000		
volsnap	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Not Available	Not Available	Not Available	ROOT\LEGACY_VOLSNAP\0000		
Remote Access IP ARP Driver	Not Available	LEGACYDRIVER	Not Available	Not Available	Not Available
Available	Not Available	Not Available	Not Available		
ROOT\LEGACY_WANARP\0000					
Audio Codecs	Yes	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMACM			
Legacy Audio Drivers	Yes	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMDRV			
Media Control Devices	Yes	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMMCI			
Legacy Video Capture Devices	Yes	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMVCD			
Video Codecs	Yes	MEDIA 5.2.3790.0	10/1/2002	(Standard system devices)	
wave.inf	Not Available	ROOT\MEDIA\MS_MMVID			
WAN Miniport (L2TP)	Yes	NET 5.2.3790.0	10/1/2002	Microsoft	
netrasa.inf	Not Available	ROOT\MS_L2TPMINIPORT\0000			
WAN Miniport (IP)	Yes	NET 5.2.3790.0	10/1/2002	Microsoft	
netrasa.inf	Not Available	ROOT\MS_NDISWANIP\0000			
WAN Miniport (PPPOE)	Yes	NET 5.2.3790.0	10/1/2002	Microsoft	
netrasa.inf	Not Available	ROOT\MS_PPPOEMINIPORT\0000			
WAN Miniport (PPTP)	Yes	NET 5.2.3790.0	10/1/2002	Microsoft	
netrasa.inf	Not Available	ROOT\MS_PPTPMINIPORT\0000			
Direct Parallel	Yes	NET 5.2.3790.0	10/1/2002	Microsoft	netrasa.inf
Not Available	Not Available	ROOT\MS_PTMINIPORT\0000			

Appendix C – Tunable Parameters

Terminal Server Device Redirector (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002
		machine.inf	Not Available	ROOT\RD\DPDR\0000
Terminal Server Keyboard Driver (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002
		machine.inf	Not Available	ROOT\RD\DP_KBD\0000
Terminal Server Mouse Driver (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002
		machine.inf	Not Available	ROOT\RD\DP_MOU\0000
Plug and Play Software Device Enumerator 10/1/2002 (Standard system devices)	Yes	SYSTEM	5.2.3790.0	
		machine.inf	Not Available	ROOT\SYSTEM\0000
Microcode Update Device (Standard system devices)	Yes	SYSTEM	5.2.3790.0	10/1/2002
		machine.inf	Not Available	ROOT\SYSTEM\0001

[Environment Variables]

Variable	Value	User Name
ComSpec	%SystemRoot%\system32\cmd.exe	<SYSTEM>
Path	C:\mksnt;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\PROGRA~1\MICROS~1\80\Tools\BINN;	<SYSTEM>
windir	%SystemRoot%	<SYSTEM>
OS	Windows_NT	<SYSTEM>
PROCESSOR_ARCHITECTURE	x86	<SYSTEM>
PROCESSOR_LEVEL	15	<SYSTEM>
PROCESSOR_IDENTIFIER	x86 Family 15 Model 2 Stepping 7, GenuineIntel	<SYSTEM>
PROCESSOR_REVISION	0207	<SYSTEM>
NUMBER_OF_PROCESSORS	4	<SYSTEM>
ClusterLog	C:\WINDOWS\Cluster\cluster.log	<SYSTEM>
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.VBS	<SYSTEM>
TEMP	%SystemRoot%\TEMP	<SYSTEM>
TMP	%SystemRoot%\TEMP	<SYSTEM>
ROOTDIR	C:/	<SYSTEM>
SHELL	C:/mksnt/sh.exe	<SYSTEM>
HOME	C:/Documents and Settings/Administrator	<SYSTEM>
TMPDIR	C:/WINDOWS/TEMP	<SYSTEM>
TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\SYSTEM
TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\LOCAL SERVICE
TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\LOCAL SERVICE
TEMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\NETWORK SERVICE
TMP	%USERPROFILE%\Local Settings\Temp	NT AUTHORITY\NETWORK SERVICE
TEMP	%USERPROFILE%\Local Settings\Temp	CLIENT77\Administrator
TMP	%USERPROFILE%\Local Settings\Temp	CLIENT77\Administrator

[Print Jobs]

Document	Size	Owner	Notify	Status	Time Submitted	Start Time	Until Time
	Elapsed Time	Pages Printed	Job ID	Priority	Parameters	Driver	Print Processor
	Host Print Queue	Data Type	Name				

[Network Connections]

Local Name	Remote Name	Type	Status	User Name
------------	-------------	------	--------	-----------

[Running Tasks]

Appendix C – Tunable Parameters

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	Not Available	Not Available
system	Not Available	4	8	0	1413120	Not Available	Not Available	Not Available	Not Available
smss.exe	Not Available	492	11	204800	1413120	1/14/2004 11:39 AM	Not Available	Not Available	Not Available
csrss.exe	Not Available	540	13	Not Available	Not Available	1/14/2004 11:39 AM	Not Available	Not Available	Not Available
winlogon.exe	c:\windows\system32\winlogon.exe	564	13	204800	1413120	1/14/2004 11:39 AM	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/29/2003 12:00 AM
services.exe	c:\windows\system32\services.exe	616	9	204800	1413120	1/14/2004 11:39 AM	5.2.3790.0 (srv03_rtm.030324-2048)	102.00 KB (104,448 bytes)	3/29/2003 12:00 AM
lsass.exe	c:\windows\system32\lsass.exe	628	9	204800	1413120	1/14/2004 11:39 AM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
svchost.exe	c:\windows\system32\svchost.exe	788	8	204800	1413120	1/14/2004 11:39 AM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
svchost.exe	c:\windows\system32\svchost.exe	844	8	204800	1413120	1/14/2004 11:39 AM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
svchost.exe	Not Available	1004	8	Not Available	Not Available	1/14/2004 11:39 AM	Not Available	Not Available	Not Available
svchost.exe	Not Available	1076	8	Not Available	Not Available	1/14/2004 11:39 AM	Not Available	Not Available	Not Available
svchost.exe	c:\windows\system32\svchost.exe	1088	8	204800	1413120	1/14/2004 11:39 AM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
spoolsv.exe	c:\windows\system32\spoolsv.exe	1312	8	204800	1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	55.00 KB (56,320 bytes)	3/29/2003 12:00 AM
msdtc.exe	Not Available	1344	8	Not Available	Not Available	1/14/2004 11:40 AM	Not Available	Not Available	Not Available
svchost.exe	c:\windows\system32\svchost.exe	1504	8	204800	1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
inetinfo.exe	c:\windows\system32\inetinfo.exe	1572	8	204800	1413120	1/14/2004 11:40 AM	6.0.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	11/13/2003 2:13 PM
svchost.exe	Not Available	1700	8	Not Available	Not Available	1/14/2004 11:40 AM	Not Available	Not Available	Not Available
dfssvc.exe	c:\windows\system32\dfssvc.exe	2036	8	204800	1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	130.50 KB (133,632 bytes)	3/29/2003 12:00 AM
svchost.exe	c:\windows\system32\svchost.exe	300	8	204800	1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	13.00 KB (13,312 bytes)	3/29/2003 12:00 AM
explorer.exe	c:\windows\explorer.exe	536	8	204800	1413120	1/14/2004 11:40 AM	6.00.3790.0 (srv03_rtm.030324-2048)	1,008.50 KB (1,032,704 bytes)	3/29/2003 12:00 AM

Appendix C – Tunable Parameters

helpctr.exe	c:\windows\pchealth\helpctr\binaries\helpctr.exe	1728	8	204800	
1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	764.00		
KB (782,336 bytes)	11/13/2003 1:32 PM				
helpsvc.exe	c:\windows\pchealth\helpctr\binaries\helpsvc.exe	1928	8	204800	
1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	720.00		
KB (737,280 bytes)	11/13/2003 1:32 PM				
helphost.exe	c:\windows\pchealth\helpctr\binaries\helphost.exe	2000	8	204800	
1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	106.00		
KB (108,544 bytes)	11/13/2003 1:32 PM				
helpctr.exe	c:\windows\pchealth\helpctr\binaries\helpctr.exe	440	8	204800	
1413120	1/14/2004 11:40 AM	5.2.3790.0 (srv03_rtm.030324-2048)	764.00		
KB (782,336 bytes)	11/13/2003 1:32 PM				
wmiprvse.exe	Not Available	480	8	Not Available	1/14/2004
11:40 AM	Not Available	Not Available	Not Available	Not Available	

[Loaded Modules]

Name	Version	Size	File Date	Manufacturer	Path
winlogon	5.2.3790.0 (srv03_rtm.030324-2048)	536.50 KB (549,376 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\winlogon.exe
ntdll	5.2.3790.0 (srv03_rtm.030324-2048)	722.50 KB (739,840 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\ntdll.dll
kernel32	5.2.3790.0 (srv03_rtm.030324-2048)	965.00 KB (988,160 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\kernel32.dll
msvcrt	7.0.3790.0 (srv03_rtm.030324-2048)	319.50 KB (327,168 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\msvcrt.dll
advapi32	5.2.3790.0 (srv03_rtm.030324-2048)	559.50 KB (572,928 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\advapi32.dll
rpcrt4	5.2.3790.0 (srv03_rtm.030324-2048)	643.50 KB (658,944 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\rpcrt4.dll
user32	5.2.3790.0 (srv03_rtm.030324-2048)	562.00 KB (575,488 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\user32.dll
gdi32	5.2.3790.0 (srv03_rtm.030324-2048)	263.00 KB (269,312 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\gdi32.dll
userenv	5.2.3790.0 (srv03_rtm.030324-2048)	732.50 KB (750,080 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\userenv.dll
nddeapi	5.2.3790.0 (srv03_rtm.030324-2048)	16.00 KB (16,384 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\nddeapi.dll
crypt32	5.131.3790.0 (srv03_rtm.030324-2048)	598.00 KB (612,352 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\crypt32.dll
msasn1	5.2.3790.0 (srv03_rtm.030324-2048)	58.00 KB (59,392 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\msasn1.dll
secur32	5.2.3790.0 (srv03_rtm.030324-2048)	63.00 KB (64,512 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\secur32.dll
winsta	5.2.3790.0 (srv03_rtm.030324-2048)	51.00 KB (52,224 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\winsta.dll
netapi32	5.2.3790.0 (srv03_rtm.030324-2048)	317.00 KB (324,608 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\netapi32.dll
profmap	5.2.3790.0 (srv03_rtm.030324-2048)	22.00 KB (22,528 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\profmap.dll
regapi	5.2.3790.0 (srv03_rtm.030324-2048)	48.50 KB (49,664 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\regapi.dll
ws2_32	5.2.3790.0 (srv03_rtm.030324-2048)	87.50 KB (89,600 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\ws2_32.dll

Appendix C – Tunable Parameters

ws2help 5.2.3790.0 (srv03_rtm.030324-2048) 19.50 KB (19,968 bytes)3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ws2help.dll
psapi 5.2.3790.0 (srv03_rtm.030324-2048) 21.50 KB (22,016 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\psapi.dll
version 5.2.3790.0 (srv03_rtm.030324-2048) 17.00 KB (17,408 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\version.dll
setupapi 5.2.3790.0 (srv03_rtm.030324-2048) 1,014.50 KB (1,038,848 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\setupapi.dll
msgina 5.2.3790.0 (srv03_rtm.030324-2048) 1.14 MB (1,191,936 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msgina.dll
shsvcs 6.00.3790.0 (srv03_rtm.030324-2048) 121.50 KB (124,416 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\shsvcs.dll
shlwapi 6.00.3790.0 (srv03_rtm.030324-2048) 281.00 KB (287,744 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\shlwapi.dll
sfc 5.2.3790.0 (srv03_rtm.030324-2048) 4.50 KB (4,608 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\sfc.dll
sfc_os 5.2.3790.0 (srv03_rtm.030324-2048) 133.00 KB (136,192 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\sfc_os.dll
wintrust 5.131.3790.0 (srv03_rtm.030324-2048) 161.50 KB (165,376 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wintrust.dll
ole32 5.2.3790.0 (srv03_rtm.030324-2048) 1.13 MB (1,187,328 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ole32.dll
imagehlp 5.2.3790.0 (srv03_rtm.030324-2048) 142.50 KB (145,920 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\imagehlp.dll
comctl32 6.0 (srv03_rtm.030324-2048) 907.00 KB (928,768 bytes) 11/13/2003
7:16 AM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-
controls_6595b64144ccf1df_6.0.100.0_x-ww_8417450b\comctl32.dll
winscard 5.2.3790.0 (srv03_rtm.030324-2048) 98.50 KB (100,864 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\winscard.dll
wtsapi32 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes)3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wtsapi32.dll
sxs 5.2.3790.0 (srv03_rtm.030324-2048) 733.00 KB (750,592 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\sxs.dll
winmm 5.2.3790.0 (srv03_rtm.030324-2048) 166.00 KB (169,984 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\winmm.dll
shell32 6.00.3790.0 (srv03_rtm.030324-2048) 7.79 MB (8,166,400 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\shell32.dll
wldap32 5.2.3790.0 (srv03_rtm.030324-2048) 158.00 KB (161,792 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\wldap32.dll
rsaenh 5.2.3790.0 (srv03_rtm.030324-2048) 176.83 KB (181,072 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\rsaenh.dll
cscdll 5.2.3790.0 (srv03_rtm.030324-2048) 99.00 KB (101,376 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\cscdll.dll
wlnotify 5.2.3790.0 (srv03_rtm.030324-2048) 87.50 KB (89,600 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wlnotify.dll
winspool 5.2.3790.0 (srv03_rtm.030324-2048) 140.00 KB (143,360 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\winspool.drv
mpr 5.2.3790.0 (srv03_rtm.030324-2048) 56.00 KB (57,344 bytes)3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\mpr.dll
comctl32 5.82 (srv03_rtm.030324-2048) 561.00 KB (574,464 bytes) 11/13/2003
7:16 AM Microsoft Corporation c:\windows\winsxs\x86_microsoft.windows.common-
controls_6595b64144ccf1df_5.82.0.0_x-ww_8a69ba05\comctl32.dll
uxtheme 6.00.3790.0 (srv03_rtm.030324-2048) 196.00 KB (200,704 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\uxtheme.dll

Appendix C – Tunable Parameters

samlib 5.2.3790.0 (srv03_rtm.030324-2048) 49.00 KB (50,176 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\samlib.dll

cscui 5.2.3790.0 (srv03_rtm.030324-2048) 305.00 KB (312,320 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\cscui.dll

oleaut32 5.2.3790.0 486.00 KB (497,664 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\oleaut32.dll

clbcatq 2001.12.4720.0 (srv03_rtm.030324-2048) 481.00 KB (492,544 bytes)
11/13/2003 1:29 PM Microsoft Corporation c:\windows\system32\clbcatq.dll

comres 2001.12.4720.0 (srv03_rtm.030324-2048) 778.00 KB (796,672 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\comres.dll

ntmarta 5.2.3790.0 (srv03_rtm.030324-2048) 114.00 KB (116,736 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ntmarta.dll

services 5.2.3790.0 (srv03_rtm.030324-2048) 102.00 KB (104,448 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\services.exe

scsvr 5.2.3790.0 (srv03_rtm.030324-2048) 316.50 KB (324,096 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\scsvr.dll

authz 5.2.3790.0 (srv03_rtm.030324-2048) 67.00 KB (68,608 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\authz.dll

umpnpgmgr 5.2.3790.0 (srv03_rtm.030324-2048) 121.50 KB (124,416 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\umpnpgmgr.dll

ncobjapi 5.2.3790.0 (srv03_rtm.030324-2048) 34.50 KB (35,328 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ncobjapi.dll

msvcp60 6.05.2144.0 388.00 KB (397,312 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\msvcp60.dll

eventlog 5.2.3790.0 (srv03_rtm.030324-2048) 60.50 KB (61,952 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\eventlog.dll

lsass 5.2.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\lsass.exe

lsasrv 5.2.3790.0 (srv03_rtm.030324-2048) 780.50 KB (799,232 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\lsasrv.dll

samsrv 5.2.3790.0 (srv03_rtm.030324-2048) 452.00 KB (462,848 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\samsrv.dll

cryptdll 5.2.3790.0 (srv03_rtm.030324-2048) 34.00 KB (34,816 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\cryptdll.dll

dnsapi 5.2.3790.0 (srv03_rtm.030324-2048) 147.50 KB (151,040 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\dnsapi.dll

ntdsapi 5.2.3790.0 (srv03_rtm.030324-2048) 76.00 KB (77,824 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\ntdsapi.dll

msprvs 5.2.3790.0 (srv03_rtm.030324-2048) 46.50 KB (47,616 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\msprvs.dll

kerberos 5.2.3790.0 (srv03_rtm.030324-2048) 332.50 KB (340,480 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\kerberos.dll

msv1_0 5.2.3790.0 (srv03_rtm.030324-2048) 127.00 KB (130,048 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msv1_0.dll

netlogon 5.2.3790.0 (srv03_rtm.030324-2048) 409.00 KB (418,816 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\netlogon.dll

w32time 5.2.3790.0 (srv03_rtm.030324-2048) 216.00 KB (221,184 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\w32time.dll

iphlpapi 5.2.3790.0 (srv03_rtm.030324-2048) 82.50 KB (84,480 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\iphlpapi.dll

schannel 5.2.3790.0 (srv03_rtm.030324-2048) 149.50 KB (153,088 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\schannel.dll

wdigest 5.2.3790.0 (srv03_rtm.030324-2048) 61.00 KB (62,464 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wdigest.dll

Appendix C – Tunable Parameters

rassfm 5.2.3790.0 (srv03_rtm.030324-2048) 20.50 KB (20,992 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rassfm.dll

kdcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 221.00 KB (226,304 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\kdcsvc.dll

ntdsa 5.2.3790.0 (srv03_rtm.030324-2048) 1.45 MB (1,520,640 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ntdsa.dll

ntdsatq 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\ntdsatq.dll

mswsock 5.2.3790.0 (srv03_rtm.030324-2048) 254.00 KB (260,096 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\mswsock.dll

esent 5.2.3790.0 (srv03_rtm.030324-2048) 1.01 MB (1,056,256 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\esent.dll

scecli 5.2.3790.0 (srv03_rtm.030324-2048) 179.50 KB (183,808 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\scecli.dll

wshtcpip 5.2.3790.0 (srv03_rtm.030324-2048) 18.00 KB (18,432 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wshtcpip.dll

ipsecsvc 5.2.3790.0 (srv03_rtm.030324-2048) 162.50 KB (166,400 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\ipsecsvc.dll

oakley 5.2.3790.0 (srv03_rtm.030324-2048) 325.50 KB (333,312 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\oakley.dll

winipsec 5.2.3790.0 (srv03_rtm.030324-2048) 34.50 KB (35,328 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\winipsec.dll

pstorsvc 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\pstorsvc.dll

psbase 5.2.3790.0 (srv03_rtm.030324-2048) 81.00 KB (82,944 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\psbase.dll

dssenh 5.2.3790.0 (srv03_rtm.030324-2048) 131.33 KB (134,480 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\dssenh.dll

wlbsctrl 5.2.3790.0 (srv03_rtm.030324-2048) 78.00 KB (79,872 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wlbsctrl.dll

w3ssl 6.0.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\w3ssl.dll

strmfilt 6.0.3790.0 (srv03_rtm.030324-2048) 70.50 KB (72,192 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\strmfilt.dll

httpapi 5.2.3790.0 (srv03_rtm.030324-2048) 26.50 KB (27,136 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\httpapi.dll

svchost 5.2.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\svchost.exe

rpcss 5.2.3790.0 (srv03_rtm.030324-2048) 276.50 KB (283,136 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\rpcss.dll

termsrv 5.2.3790.0 (srv03_rtm.030324-2048) 216.50 KB (221,696 bytes) 11/13/2003
1:29 PM Microsoft Corporation c:\windows\system32\termsrv.dll

icaapi 5.2.3790.0 (srv03_rtm.030324-2048) 10.50 KB (10,752 bytes) 11/13/2003 1:29 PM
Microsoft Corporation c:\windows\system32\icaapi.dll

mstlsapi 5.2.3790.0 (srv03_rtm.030324-2048) 104.50 KB (107,008 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\mstlsapi.dll

activeds 5.2.3790.0 (srv03_rtm.030324-2048) 189.00 KB (193,536 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\activeds.dll

adslidpc 5.2.3790.0 (srv03_rtm.030324-2048) 142.50 KB (145,920 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\adslidpc.dll

credui 5.2.3790.0 (srv03_rtm.030324-2048) 159.00 KB (162,816 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\credui.dll

atl 3.05.2283 83.00 KB (84,992 bytes) 3/29/2003 12:00 AM Microsoft Corporation
c:\windows\system32\atl.dll

Appendix C – Tunable Parameters

rdpwsx 5.2.3790.0 (srv03_rtm.030324-2048) 80.13 KB (82,056 bytes) 11/13/2003 1:29 PM
Microsoft Corporation c:\windows\system32\rdpwsx.dll

wzcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 272.50 KB (279,040 bytes) 3/25/2003 6:15 AM
Microsoft Corporation c:\windows\system32\wzcsvc.dll

rtutils 5.2.3790.0 (srv03_rtm.030324-2048) 32.00 KB (32,768 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rtutils.dll

wmi 5.2.3790.0 (srv03_rtm.030324-2048) 6.50 KB (6,656 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wmi.dll

dhcpcsvc 5.2.3790.0 (srv03_rtm.030324-2048) 101.50 KB (103,936 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\dhcpcsvc.dll

rastls 5.2.3790.0 (srv03_rtm.030324-2048) 155.00 KB (158,720 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rastls.dll

cryptui 5.131.3790.0 (srv03_rtm.030324-2048) 473.50 KB (484,864 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\cryptui.dll

mprapi 5.2.3790.0 (srv03_rtm.030324-2048) 81.00 KB (82,944 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\mprapi.dll

rasapi32 5.2.3790.0 (srv03_rtm.030324-2048) 227.50 KB (232,960 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rasapi32.dll

rasman 5.2.3790.0 (srv03_rtm.030324-2048) 56.50 KB (57,856 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\rasman.dll

tapi32 5.2.3790.0 (srv03_rtm.030324-2048) 175.00 KB (179,200 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\tapi32.dll

raschap 5.2.3790.0 (srv03_rtm.030324-2048) 106.00 KB (108,544 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\raschap.dll

schedsvc 5.2.3790.0 (srv03_rtm.030324-2048) 176.00 KB (180,224 bytes) 11/13/2003 1:32 PM
Microsoft Corporation c:\windows\system32\schedsvc.dll

wiarpc 5.2.3790.0 (srv03_rtm.030324-2048) 30.00 KB (30,720 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wiarpc.dll

msidle 6.00.3790.0 (srv03_rtm.030324-2048) 5.50 KB (5,632 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\msidle.dll

audiosrv 5.2.3790.0 (srv03_rtm.030324-2048) 38.00 KB (38,912 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\audiosrv.dll

wkssvc 5.2.3790.0 (srv03_rtm.030324-2048) 125.00 KB (128,000 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\wkssvc.dll

cryptsvc 5.2.3790.0 (srv03_rtm.030324-2048) 51.00 KB (52,224 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\cryptsvc.dll

certcli 5.2.3790.0 (srv03_rtm.030324-2048) 228.00 KB (233,472 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\certcli.dll

vssapi 5.2.3790.0 (srv03_rtm.030324-2048) 528.00 KB (540,672 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\vssapi.dll

dmserver 5.2.3790.0 (srv03_rtm.030324-2048) 24.00 KB (24,576 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\dmserver.dll

es 2001.12.4720.0 (srv03_rtm.030324-2048) 221.50 KB (226,816 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\es.dll

pchsvc 5.2.3790.0 (srv03_rtm.030324-2048) 31.50 KB (32,256 bytes) 11/13/2003 1:32 PM
Microsoft Corporation c:\windows\pchealth\helpctr\binaries\pchsvc.dll

srvsvc 5.2.3790.0 (srv03_rtm.030324-2048) 89.00 KB (91,136 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\srvsvc.dll

seclogon 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\seclogon.dll

sens 5.2.3790.0 (srv03_rtm.030324-2048) 35.50 KB (36,352 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\sens.dll

trkwks 5.2.3790.0 (srv03_rtm.030324-2048) 85.00 KB (87,040 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\trkwks.dll

Appendix C – Tunable Parameters

wmisvc 5.2.3790.0 (srv03_rtm.030324-2048) 131.00 KB (134,144 bytes) 11/13/2003
1:29 PM Microsoft Corporation c:\windows\system32\wbem\wmisvc.dll

wuauerv 5.4.3790.0 (srv03_rtm.030324-2048) 10.50 KB (10,752 bytes) 11/13/2003
1:29 PM Microsoft Corporation c:\windows\system32\wuauerv.dll

wuaueng 5.4.3790.0 (srv03_rtm.030324-2048) 188.50 KB (193,024 bytes)
11/13/2003 1:29 PM Microsoft Corporation c:\windows\system32\wuaueng.dll

advpack 6.00.3790.0 (srv03_rtm.030324-2048) 93.50 KB (95,744 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\advpack.dll

wininet 6.00.3790.0 (srv03_rtm.030324-2048) 609.00 KB (623,616 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wininet.dll

winrntr 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\winrntr.dll

comsvcs 2001.12.4720.0 (srv03_rtm.030324-2048) 1.14 MB (1,199,616 bytes)
11/13/2003 1:29 PM Microsoft Corporation c:\windows\system32\comsvcs.dll

browser5.2.3790.0 (srv03_rtm.030324-2048) 70.50 KB (72,192 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\browser.dll

actxprxy 6.00.3790.0 (srv03_rtm.030324-2048) 95.00 KB (97,280 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\actxprxy.dll

netrap 5.2.3790.0 (srv03_rtm.030324-2048) 11.50 KB (11,776 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\netrap.dll

rasadhlp 5.2.3790.0 (srv03_rtm.030324-2048) 6.50 KB (6,656 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\rasadhlp.dll

netman 5.2.3790.0 (srv03_rtm.030324-2048) 209.00 KB (214,016 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\netman.dll

wzcsapi5.2.3790.0 (srv03_rtm.030324-2048) 24.50 KB (25,088 bytes) 3/25/2003 6:15 AM
Microsoft Corporation c:\windows\system32\wzcsapi.dll

netshell 5.2.3790.0 (srv03_rtm.030324-2048) 1.67 MB (1,747,456 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\netshell.dll

clusapi 5.2.3790.0 (srv03_rtm.030324-2048) 56.00 KB (57,344 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\clusapi.dll

netcfgx 5.2.3790.0 (srv03_rtm.030324-2048) 726.00 KB (743,424 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\netcfgx.dll

hnetcfg 5.2.3790.0 (srv03_rtm.030324-2048) 243.50 KB (249,344 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\hnetcfg.dll

wbemprox 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes) 11/13/2003
1:29 PM Microsoft Corporation c:\windows\system32\wbem\wbemprox.dll

wbemcomn 5.2.3790.0 (srv03_rtm.030324-2048) 211.50 KB (216,576 bytes)
3/29/2003 12:00 AM Microsoft Corporation
c:\windows\system32\wbem\wbemcomn.dll

wbemcore 5.2.3790.0 (srv03_rtm.030324-2048) 457.00 KB (467,968 bytes)
11/13/2003 1:29 PM Microsoft Corporation
c:\windows\system32\wbem\wbemcore.dll

esscli 5.2.3790.0 (srv03_rtm.030324-2048) 235.50 KB (241,152 bytes) 11/13/2003
1:29 PM Microsoft Corporation c:\windows\system32\wbem\esscli.dll

fastprox5.2.3790.0 (srv03_rtm.030324-2048) 443.00 KB (453,632 bytes) 11/13/2003
1:29 PM Microsoft Corporation c:\windows\system32\wbem\fastprox.dll

wbemsvc 5.2.3790.0 (srv03_rtm.030324-2048) 42.50 KB (43,520 bytes) 11/13/2003
1:29 PM Microsoft Corporation c:\windows\system32\wbem\wbemsvc.dll

wmiutils5.2.3790.0 (srv03_rtm.030324-2048) 90.50 KB (92,672 bytes) 11/13/2003 1:29 PM
Microsoft Corporation c:\windows\system32\wbem\wmiutils.dll

repdrvfs 5.2.3790.0 (srv03_rtm.030324-2048) 165.00 KB (168,960 bytes)
11/13/2003 1:29 PM Microsoft Corporation c:\windows\system32\wbem\repdrvfs.dll

wmiprvsd 5.2.3790.0 (srv03_rtm.030324-2048) 405.50 KB (415,232 bytes)
11/13/2003 1:29 PM Microsoft Corporation
c:\windows\system32\wbem\wmiprvsd.dll

Appendix C – Tunable Parameters

wbemess 5.2.3790.0 (srv03_rtm.030324-2048) 256.50 KB (262,656 bytes)
11/13/2003 1:29 PM Microsoft Corporation
c:\windows\system32\wbem\wbemess.dll

rasdlg 5.2.3790.0 (srv03_rtm.030324-2048) 642.00 KB (657,408 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\rasdlg.dll

spoolsv 5.2.3790.0 (srv03_rtm.030324-2048) 55.00 KB (56,320 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\spoolsv.exe

spoolss 5.2.3790.0 (srv03_rtm.030324-2048) 79.00 KB (80,896 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\spoolss.dll

localspl 5.2.3790.0 (srv03_rtm.030324-2048) 304.50 KB (311,808 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\localspl.dll

cnbjmon 5.2.3680.0 (Lab03_dev(skatar).020509-1043) 45.50 KB (46,592 bytes)
3/24/2003 7:48 PM Microsoft Corporation c:\windows\system32\cnbjmon.dll

pjlmon 5.2.3790.0 (srv03_rtm.030324-2048) 15.00 KB (15,360 bytes) 3/24/2003 7:49 PM
Microsoft Corporation c:\windows\system32\pjlmon.dll

tcpmon 5.2.3790.0 (srv03_rtm.030324-2048) 44.00 KB (45,056 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\tcpmon.dll

mgmtapi 5.2.3790.0 (srv03_rtm.030324-2048) 14.00 KB (14,336 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\mgmtapi.dll

snmpapi 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\snmpapi.dll

wsnmp32 5.2.3790.0 (srv03_rtm.030324-2048) 39.50 KB (40,448 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wsnmp32.dll

usbmon 5.2.3790.0 (srv03_rtm.030324-2048) 17.00 KB (17,408 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\usbmon.dll

win32spl 5.2.3790.0 (srv03_rtm.030324-2048) 94.50 KB (96,768 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\win32spl.dll

inetpp 5.2.3790.0 (srv03_rtm.030324-2048) 71.50 KB (73,216 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\inetpp.dll

icmp 5.2.3790.0 (srv03_rtm.030324-2048) 4.50 KB (4,608 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\icmp.dll

ersvc 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\ersvc.dll

inetinfo 6.0.3790.0 (srv03_rtm.030324-2048) 13.00 KB (13,312 bytes) 11/13/2003 2:13 PM
Microsoft Corporation c:\windows\system32\inetsrv\inetinfo.exe

iisutil 6.0.3790.0 (srv03_rtm.030324-2048) 177.00 KB (181,248 bytes) 11/13/2003
2:13 PM Microsoft Corporation c:\windows\system32\inetsrv\iisutil.dll

rpcref 6.0.3790.0 (srv03_rtm.030324-2048) 4.00 KB (4,096 bytes) 11/13/2003 2:13 PM
Microsoft Corporation c:\windows\system32\inetsrv\rpcref.dll

iisrtl 6.0.3790.0 (srv03_rtm.030324-2048) 129.00 KB (132,096 bytes) 11/13/2003
2:13 PM Microsoft Corporation c:\windows\system32\iisrtl.dll

iisadmin 6.0.3790.0 (srv03_rtm.030324-2048) 18.50 KB (18,944 bytes) 11/13/2003
2:13 PM Microsoft Corporation c:\windows\system32\inetsrv\iisadmin.dll

coadmin 6.0.3790.0 (srv03_rtm.030324-2048) 48.50 KB (49,664 bytes) 11/13/2003
2:13 PM Microsoft Corporation c:\windows\system32\inetsrv\coadmin.dll

admwprox 6.0.3790.0 (srv03_rtm.030324-2048) 44.00 KB (45,056 bytes) 11/13/2003
2:13 PM Microsoft Corporation c:\windows\system32\admwprox.dll

iiscfg 6.0.3790.0 (srv03_rtm.030324-2048) 1.06 MB (1,116,160 bytes) 11/13/2003
2:13 PM Microsoft Corporation c:\windows\system32\inetsrv\iiscfg.dll

metadata 6.0.3790.0 (srv03_rtm.030324-2048) 218.50 KB (223,744 bytes)
11/13/2003 2:13 PM Microsoft Corporation
c:\windows\system32\inetsrv\metadata.dll

msxml3 8.40.9419.0 1.28 MB (1,337,344 bytes) 3/29/2003 12:00 AM Microsoft
Corporation c:\windows\system32\msxml3.dll

Appendix C – Tunable Parameters

svcext 6.0.3790.0 (srv03_rtm.030324-2048) 41.50 KB (42,496 bytes) 11/13/2003 2:13 PM
Microsoft Corporation c:\windows\system32\inetsrv\svcext.dll

security 5.2.3790.0 (srv03_rtm.030324-2048) 5.50 KB (5,632 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\security.dll

iismap 6.0.3790.0 (srv03_rtm.030324-2048) 55.00 KB (56,320 bytes) 11/13/2003 2:13 PM
Microsoft Corporation c:\windows\system32\iismap.dll

wamreg 6.0.3790.0 (srv03_rtm.030324-2048) 52.00 KB (53,248 bytes) 11/13/2003 2:13 PM
Microsoft Corporation c:\windows\system32\inetsrv\wamreg.dll

dfssvc 5.2.3790.0 (srv03_rtm.030324-2048) 130.50 KB (133,632 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\dfssvc.exe

resutils 5.2.3790.0 (srv03_rtm.030324-2048) 59.00 KB (60,416 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\resutils.dll

mfc42u 6.05.3014.0 960.00 KB (983,040 bytes) 3/29/2003 12:00 AM Microsoft
Corporation c:\windows\system32\mfc42u.dll

wsock32 5.2.3790.0 (srv03_rtm.030324-2048) 22.00 KB (22,528 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\wsock32.dll

iisw3adm 6.0.3790.0 (srv03_rtm.030324-2048) 199.50 KB (204,288 bytes)
11/13/2003 2:13 PM Microsoft Corporation
c:\windows\system32\inetsrv\iisw3adm.dll

w3cache 6.0.3790.0 (srv03_rtm.030324-2048) 21.00 KB (21,504 bytes) 11/13/2003
2:13 PM Microsoft Corporation c:\windows\system32\inetsrv\w3cache.dll

w3tp 6.0.3790.0 (srv03_rtm.030324-2048) 12.50 KB (12,800 bytes) 11/13/2003 2:13 PM
Microsoft Corporation c:\windows\system32\inetsrv\w3tp.dll

lonsint 6.0.3790.0 (srv03_rtm.030324-2048) 11.50 KB (11,776 bytes) 11/13/2003 2:13 PM
Microsoft Corporation c:\windows\system32\inetsrv\lonsint.dll

explorer 6.00.3790.0 (srv03_rtm.030324-2048) 1,008.50 KB (1,032,704 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\explorer.exe

browseui 6.00.3790.0 (srv03_rtm.030324-2048) 1.01 MB (1,057,280 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\browseui.dll

shdocvw 6.00.3790.0 (srv03_rtm.030324-2048) 1.33 MB (1,393,664 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\shdocvw.dll

apphelp 5.2.3790.0 (srv03_rtm.030324-2048) 122.00 KB (124,928 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\apphelp.dll

themeui 6.00.3790.0 (srv03_rtm.030324-2048) 360.50 KB (369,152 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\themeui.dll

msimg32 5.2.3790.0 (srv03_rtm.030324-2048) 4.50 KB (4,608 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\msimg32.dll

linkinfo 5.2.3790.0 (srv03_rtm.030324-2048) 16.50 KB (16,896 bytes) 3/29/2003 12:00 AM
Microsoft Corporation c:\windows\system32\linkinfo.dll

ntshrui 6.00.3790.0 (srv03_rtm.030324-2048) 136.00 KB (139,264 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\ntshrui.dll

webcheck 6.00.3790.0 (srv03_rtm.030324-2048) 261.50 KB (267,776 bytes)
3/29/2003 12:00 AM Microsoft Corporation c:\windows\system32\webcheck.dll

stobject 5.2.3790.0 (srv03_rtm.030324-2048) 117.50 KB (120,320 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\stobject.dll

batmeter 6.00.3790.0 (srv03_rtm.030324-2048) 28.50 KB (29,184 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\batmeter.dll

powrprof 6.00.3790.0 (srv03_rtm.030324-2048) 14.50 KB (14,848 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\powrprof.dll

printui 5.2.3790.0 (srv03_rtm.030324-2048) 536.50 KB (549,376 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\printui.dll

cfgmgr32 5.2.3790.0 (srv03_rtm.030324-2048) 17.50 KB (17,920 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\cfgmgr32.dll

urlmon 6.00.3790.0 (srv03_rtm.030324-2048) 501.50 KB (513,536 bytes) 3/29/2003
12:00 AM Microsoft Corporation c:\windows\system32\urlmon.dll

Appendix C – Tunable Parameters

helpctr	5.2.3790.0 (srv03_rtm.030324-2048)	764.00 KB (782,336 bytes)	11/13/2003 1:32 PM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\helpctr.exe
hcappres	5.2.3790.0 (srv03_rtm.030324-2048)	6.50 KB (6,656 bytes)	11/13/2003 1:32 PM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\hcappres.dll
itss	5.2.3790.0 (srv03_rtm.030324-2048)	119.50 KB (122,368 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\itss.dll
pchshell	5.2.3790.0 (srv03_rtm.030324-2048)	100.50 KB (102,912 bytes)	11/13/2003 1:32 PM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\pchshell.dll
mlang	6.00.3790.0 (srv03_rtm.030324-2048)	570.00 KB (583,680 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\mlang.dll
mshtml	6.00.3790.0 (srv03_rtm.030324-2048)	2.78 MB (2,916,352 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\mshtml.dll
msimtf	5.2.3790.0 (srv03_rtm.030324-2048)	149.00 KB (152,576 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\msimtf.dll
msctf	5.2.3790.0 (srv03_rtm.030324-2048)	287.00 KB (293,888 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\msctf.dll
shdoclc	6.00.3790.0 (srv03_rtm.030324-2048)	588.50 KB (602,624 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\shdoclc.dll
jscrip	5.6.0.8515	436.00 KB (446,464 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\jscrip.dll
msls31	3.10.349.0	147.00 KB (150,528 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\msls31.dll
imm32	5.2.3790.0 (srv03_rtm.030324-2048)	105.50 KB (108,032 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\imm32.dll
mshtml	6.00.3790.0 (srv03_rtm.030324-2048)	443.50 KB (454,144 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\mshtml.dll
imgutil	5.2.3790.0 (srv03_rtm.030324-2048)	35.00 KB (35,840 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\imgutil.dll
wbemdisp	5.2.3790.0 (srv03_rtm.030324-2048)	165.50 KB (169,472 bytes)	11/13/2003 1:29 PM	Microsoft Corporation	c:\windows\system32\wbem\wbemdisp.dll
wshom	5.6.0.8515	92.00 KB (94,208 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\wshom.ocx
scrrun	5.6.0.8515	148.00 KB (151,552 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\scrrun.dll
mfc42	6.05.3014.0	960.00 KB (983,040 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\mfc42.dll
helpsvc	5.2.3790.0 (srv03_rtm.030324-2048)	720.00 KB (737,280 bytes)	11/13/2003 1:32 PM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\helpsvc.exe
helpost	5.2.3790.0 (srv03_rtm.030324-2048)	106.00 KB (108,544 bytes)	11/13/2003 1:32 PM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\helpost.exe
sensapi	5.2.3790.0 (srv03_rtm.030324-2048)	6.00 KB (6,144 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\sensapi.dll
vbscript	5.6.0.8515	404.00 KB (413,696 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\vbscript.dll
msinfo	5.2.3790.0 (srv03_rtm.030324-2048)	358.50 KB (367,104 bytes)	11/13/2003 1:32 PM	Microsoft Corporation	c:\windows\pchealth\helpctr\binaries\msinfo.dll
comdlg32	6.00.3790.0 (srv03_rtm.030324-2048)	261.00 KB (267,264 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\comdlg32.dll
riched32	5.2.3790.0 (srv03_rtm.030324-2048)	3.50 KB (3,584 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\riched32.dll
riched20	5.31.23.1218	406.00 KB (415,744 bytes)	3/29/2003 12:00 AM	Microsoft Corporation	c:\windows\system32\riched20.dll

Appendix C – Tunable Parameters

[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	Running		Auto	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	Running		Auto	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\dlhhost.exe /processid:{02d4b3f1-fd88-11d1-960d-00805fc79235}		Normal	LocalSystem			0
Cryptographic Services	CryptSvc	Running		Auto	Share Process		
	c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem			0
Distributed File System	Dfs	Running		Auto	Own Process		
	c:\windows\system32\dfsrv.exe		Normal	LocalSystem			0
DHCP Client	Dhcp	Running		Auto	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\dmadmin.exe /com		Normal	LocalSystem			0
Logical Disk Manager	dmserver	Running		Auto	Share Process		
	c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem			0
DNS Client	Dnscache	Running		Auto	Share Process		
	c:\windows\system32\svchost.exe -k networkservice		Normal	NT			
AUTHORITY\NetworkService			0				
Error Reporting Service	ERSvc	Running		Auto	Share Process		
	c:\windows\system32\svchost.exe -k 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		Auto	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	Running		Manual	Share Process		
	c:\windows\system32\lsass.exe		Normal	LocalSystem			0
IIS Admin Service	IISADMIN	Running		Auto	Share Process		
	c:\windows\system32\inetrv\inetinfo.exe		Normal	LocalSystem			0
IMAPI CD-Burning COM Service	ImapiService	Stopped			Disabled	Own	
Process	c:\windows\system32\imapi.exe		Normal	LocalSystem			0

Appendix C – Tunable Parameters

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	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
Workstation	lanmanworkstation	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
License Logging	LicenseService	Stopped	Disabled	Own Process	
	c:\windows\system32\lssrv.exe	Normal NT AUTHORITY\NetworkService		0	
TCP/IP NetBIOS Helper	LmHosts	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k localservice		Normal NT		
AUTHORITY\LocalService		0			
Messenger	Messenger	Stopped	Disabled	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
NetMeeting Remote Desktop Sharing	mnmsrvc	Stopped	Disabled	Own	
Process	c:\windows\system32\mnmsrvc.exe		Normal LocalSystem	0	
Distributed Transaction Coordinator	MSDTC	Running	Auto	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	
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	
	c:\windows\system32\lsass.exe	Normal LocalSystem	0		
Removable Storage	NtmsSvc	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
Plug and Play	PlugPlay	Running	Auto	Share Process	
	c:\windows\system32\services.exe		Normal LocalSystem	0	
IPSEC Services	PolicyAgent	Running	Auto	Share Process	
	c:\windows\system32\lsass.exe	Normal LocalSystem	0		
Protected Storage	ProtectedStorage	Running	Auto	Share Process	
	c:\windows\system32\lsass.exe	Normal LocalSystem	0		
Remote Access Auto Connection Manager	RasAuto	Stopped	Manual	Share Process	
Process	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
Remote Access Connection Manager	RasMan	Stopped	Manual	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
Remote Desktop Help Session Manager	RDSessMgr	Stopped	Manual	Own Process	
	c:\windows\system32\sessmgr.exe		Normal LocalSystem	0	
Routing and Remote Access	RemoteAccess	Stopped	Disabled	Share Process	
	c:\windows\system32\svchost.exe -k netsvcs		Normal LocalSystem	0	
Remote Registry	RemoteRegistry	Running	Auto	Share Process	
	c:\windows\system32\svchost.exe -k regsvc		Normal NT AUTHORITY\LocalService	0	

Appendix C – Tunable Parameters

Remote Procedure Call (RPC) Locator	RpcLocator	Stopped	Manual	Own Process
c:\windows\system32\locator.exe		Normal	NT AUTHORITY\NetworkService	0
Remote Procedure Call (RPC)	RpcSs	Running	Auto	Share Process
c:\windows\system32\svchost -k rpcss		Normal	LocalSystem	0
Resultant Set of Policy Provider	RSOPProv	Stopped	Manual	Share Process
c:\windows\system32\rsopprov.exe		Normal	LocalSystem	0
Special Administration Console Helper	sacsrv	Stopped	Manual	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Security Accounts Manager	SamSs	Running	Auto	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		Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Secondary Logon	seclogon	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Ignore	LocalSystem	0
System Event Notification	SENS	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Internet Connection Firewall (ICF) / Internet Connection Sharing (ICS)		Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Shell Hardware Detection	ShellHWDetection	Running	Auto	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Ignore	LocalSystem	0
Print Spooler	Spooler	Running	Auto	Own Process
c:\windows\system32\spoolsv.exe		Normal	LocalSystem	0
Windows Image Acquisition (WIA)	stisvc	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k imgsvc		Normal	NT AUTHORITY\LocalService	0
Microsoft Software Shadow Copy Provider	swprv	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 termsvcs		Normal	LocalSystem	0
Themes	Themes	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Telnet	TlntSvr	Stopped	Disabled	Share Process
c:\windows\system32\tlntsvr.exe		Normal	NT AUTHORITY\LocalService	0
Distributed Link Tracking Server	TrkSvr	Stopped	Disabled	Share Process
c:\windows\system32\svchost.exe -k netsvcs		Normal	LocalSystem	0
Distributed Link Tracking Client	TrkWks	Running	Auto	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

Appendix C – Tunable Parameters

```

Windows Time W32Time Running Auto Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
World Wide Web Publishing Service W3SVCRunning Auto Share Process
c:\windows\system32\svchost.exe -k iissvcs 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 wuauclt Running Auto Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0
Wireless Configuration WZCSVC Running Auto Share Process
c:\windows\system32\svchost.exe -k netsvcs Normal LocalSystem 0

```

[Program Groups]

```

Group Name Name User Name
Accessories Default User:Accessories Default User
Accessories\Accessibility Default User:Accessories\Accessibility Default User
Accessories\Entertainment Default User:Accessories\Entertainment Default User
Startup Default User:Startup Default User
Accessories All Users:Accessories All Users
Accessories\Accessibility All Users:Accessories\Accessibility All Users
Accessories\Communications All Users:Accessories\Communications All Users
Accessories\Entertainment All Users:Accessories\Entertainment All Users
Accessories\System Tools All Users:Accessories\System Tools All Users
Administrative Tools All Users:Administrative Tools All Users
Microsoft SQL Server All Users:Microsoft SQL Server All Users
MKS Toolkit All Users:MKS Toolkit 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 CLIENT77\Administrator:Accessories CLIENT77\Administrator
Accessories\Accessibility CLIENT77\Administrator:Accessories\Accessibility
CLIENT77\Administrator
Accessories\Entertainment CLIENT77\Administrator:Accessories\Entertainment
CLIENT77\Administrator
Administrative Tools CLIENT77\Administrator:Administrative Tools
CLIENT77\Administrator
Startup CLIENT77\Administrator:StartupCLIENT77\Administrator

```


Appendix C – Tunable Parameters

[Startup Programs]

Program	Command	User Name	Location
desktop\desktop.ini		NT AUTHORITY\SYSTEM	Startup
desktop\desktop.ini		CLIENT77\Administrator	Startup
desktop\desktop.ini		.DEFAULT	Startup
desktop\desktop.ini		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
actxprxy.dll	6.0.3790.0	95 KB	3/29/2003		C:\WINDOWS\system32
					Microsoft Corporation
advpack.dll	6.0.3790.0	94 KB	3/29/2003		C:\WINDOWS\system32
					Microsoft Corporation
asctrls.ocx	6.0.3790.0	90 KB	3/29/2003		C:\WINDOWS\system32
					Microsoft Corporation

Appendix C – Tunable Parameters

browseic.dll	6.0.3790.0 Microsoft Corporation	62 KB	3/29/2003	C:\WINDOWS\system32	
browseui.dll	6.0.3790.0 Microsoft Corporation	1,033 KB	3/29/2003	C:\WINDOWS\system32	
cdfview.dll	6.0.3790.0 Microsoft Corporation	144 KB	3/29/2003	C:\WINDOWS\system32	
comctl32.dll	5.82.3790.0 Microsoft Corporation	561 KB	3/29/2003	C:\WINDOWS\system32	
dxttrans.dll	6.3.3790.0 Microsoft Corporation	198 KB	3/29/2003	C:\WINDOWS\system32	
dxtmsft.dll	6.3.3790.0 Microsoft Corporation	344 KB	3/29/2003	C:\WINDOWS\system32	
iecont.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iecontlc.dll	<File Missing>	Not Available	Not Available	Not Available	Not Available
iedkcs32.dll	16.0.3790.0 Microsoft Corporation	300 KB	3/29/2003	C:\WINDOWS\system32	
iepeers.dll	6.0.3790.0 Microsoft Corporation	230 KB	3/29/2003	C:\WINDOWS\system32	
iesetup.dll	6.0.3790.0 Microsoft Corporation	59 KB	3/29/2003	C:\WINDOWS\system32	
ieuinit.inf	Not Available Available	20 KB	3/29/2003	C:\WINDOWS\system32	Not
ieexplore.exe	6.0.3790.0 Microsoft Corporation	90 KB	3/29/2003	C:\Program Files\Internet Explorer	
imgutil.dll	5.2.3790.0 Microsoft Corporation	35 KB	3/29/2003	C:\WINDOWS\system32	
inetctl.cpl	6.0.3790.0 Microsoft Corporation	303 KB	3/29/2003	C:\WINDOWS\system32	
inetctlc.dll	6.0.3790.0 Microsoft Corporation	109 KB	3/29/2003	C:\WINDOWS\system32	
inseng.dll	6.0.3790.0 Microsoft Corporation	72 KB	3/29/2003	C:\WINDOWS\system32	
mlang.dll	6.0.3790.0 Microsoft Corporation	570 KB	3/29/2003	C:\WINDOWS\system32	
msencode.dll	2002.10.4.0 Available	112 KB	3/29/2003	C:\WINDOWS\system32	Not
mshta.exe	6.0.3790.0 Microsoft Corporation	26 KB	3/29/2003	C:\WINDOWS\system32	
mshtml.dll	6.0.3790.0 Microsoft Corporation	2,848 KB	3/29/2003	C:\WINDOWS\system32	
mshtml.tlb	6.0.3790.0 Microsoft Corporation	1,319 KB	3/29/2003	C:\WINDOWS\system32	
mshtmlled.dll	6.0.3790.0 Microsoft Corporation	444 KB	3/29/2003	C:\WINDOWS\system32	
mshtmlmer.dll	6.0.3790.0 Microsoft Corporation	55 KB	3/29/2003	C:\WINDOWS\system32	
msident.dll	6.0.3790.0 Microsoft Corporation	47 KB	3/29/2003	C:\WINDOWS\system32	
msidntld.dll	6.0.3790.0 Microsoft Corporation	15 KB	3/29/2003	C:\WINDOWS\system32	
msieftp.dll	6.0.3790.0 Microsoft Corporation	230 KB	3/29/2003	C:\WINDOWS\system32	
msrating.dll	6.0.3790.0 Microsoft Corporation	132 KB	3/29/2003	C:\WINDOWS\system32	

Appendix C – Tunable Parameters

mstime.dll	6.0.3790.0	491 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
occache.dll	6.0.3790.0	89 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
proctexe.ocx	6.3.3790.0	78 KB	3/29/2003	C:\WINDOWS\system32	Intel Corporation
sendmail.dll	6.0.3790.0	52 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
shdoclc.dll	6.0.3790.0	589 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
shdocvw.dll	6.0.3790.0	1,361 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
shfolder.dll	6.0.3790.0	23 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
shlwapi.dll	6.0.3790.0	281 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
tdc.ocx	1.3.0.3130	58 KB	3/29/2003	C:\WINDOWS\system32	Microsoft Corporation
url.dll	6.0.3790.0	36 KB	3/29/2003	C:\WINDOWS\system32	Microsoft Corporation
urlmon.dll	6.0.3790.0	502 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
webcheck.dll	6.0.3790.0	262 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				
wininet.dll	6.0.3790.0	609 KB	3/29/2003	C:\WINDOWS\system32	
	Microsoft Corporation				

[Connectivity]

Item Value
Connection Preference Never dial

LAN Settings

AutoConfigProxy Not Available
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\NetworkService\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

Appendix C – Tunable Parameters

[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 AdvisorDisabled

[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

COM+ Settings

TPCC.AITxns:

 Activation:

 Enable Object Pooling selected
 Minimum Pool Size: 300
 Maximum Pool Size: 300
 Creation Timeout: 60,000
 Enable Object Construction
 Enable Just in Time Activation

 Concurrency:

 Concurrency Required

Appendix C – Tunable Parameters

TPCC Application Registry Parameters

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\TPCC]

"Path"="C:\\Inetpub\\wwwroot\\"

"NumberOfDeliveryThreads"=dword:0000004b

"MaxConnections"=dword:000061a8

"MaxPendingDeliveries"=dword:0000012c

"DB_Protocol"="ODBC"

"TxnMonitor"="COM"

"DbServer"="pe2650"

"DbName"="tpcc"

"DbUser"="sa"

"DbPassword"=""

"COM_SinglePool"="YES"

Microsoft Internet Information Server Registry Parameters

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Parameters]

"ListenBackLog"=dword:00000019

"DispatchEntries"=hex(7):4c,00,44,00,41,00,50,00,53,00,56,00,43,00,00,00,00,00

"PoolThreadLimit"=dword:000000be

"ThreadTimeout"=dword:00015180

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\InetInfo\Performance]

"Library"="infectrs.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:de,fc,ed,18,0a,98,c0,01,10,25,00,00,00,00,00,00

"WbemAdapFileTime"=hex:00,60,4e,96,aa,40,bf,01

"WbemAdapFileSize"=dword:00002510

"WbemAdapStatus"=dword:00000000

World Wide Web Service Registry Parameters

Windows Registry Editor Version 5.00

Appendix C – Tunable Parameters

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):25,00,53,00,79,00,73,00,74,00,65,00,6d,00,52,00,6f,00,6f,00,\
  74,00,25,00,5c,00,53,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,73,\
  00,76,00,63,00,68,00,6f,00,73,00,74,00,2e,00,65,00,78,00,65,00,20,00,2d,00,\
  6b,00,20,00,69,00,69,00,73,00,73,00,76,00,63,00,73,00,00,00
"DisplayName"="World Wide Web Publishing Service"
"DependOnService"=hex(7):52,00,50,00,43,00,53,00,53,00,00,00,48,00,54,00,54,00,\
  50,00,46,00,69,00,6c,00,74,00,65,00,72,00,00,00,49,00,49,00,53,00,41,00,44,\
  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 Manager"
"FailureActions"=hex:80,51,01,00,01,00,00,00,00,00,00,00,03,00,00,00,53,00,65,\
  00,01,00,00,00,01,00,00,00,01,00,00,00,01,00,00,00,01,00,00,00,01,00,00,00
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters]
"MajorVersion"=dword:00000006
"MinorVersion"=dword:00000000
"InstallPath"="C:\\WINDOWS\\system32\\inetsrv"
"AccessDeniedMessage"="Error: Access is Denied."
"ServiceDll"=hex(2):43,00,3a,00,5c,00,57,00,49,00,4e,00,44,00,4f,00,57,00,53,\
  00,5c,00,73,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,69,00,6e,00,\
  65,00,74,00,73,00,72,00,76,00,5c,00,69,00,69,00,73,00,77,00,33,00,61,00,64,\
  00,6d,00,2e,00,64,00,6c,00,6c,00,00,00
"AcceptExOutstanding"=dword:00000028
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\ADCLaun
ch]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\ADCLaun
ch\AdvancedDataFactory]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\ADCLaun
ch\RDSServer.DataFactory]
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Parameters\Virtual
Roots]
"/"="c:\\inetpub\\wwwroot,,205"
"/Scripts"="c:\\inetpub\\scripts,,204"
"/IISHelp"="c:\\winnt\\help\\iishelp,,201"
"/IISAdmin"="C:\\WINNT\\System32\\inetsrv\\iisadmin,,201"
"/IISSamples"="c:\\inetpub\\iissamples,,201"
"/MSADC"="c:\\program files\\common files\\system\\msadc,,205"
"/_vti_bin"="C:\\Program Files\\Common Files\\Microsoft Shared\\Web Server
Extensions\\40\\isapi,,205"
"/Printers"="C:\\WINNT\\web\\printers,,201"
```

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Performance]
"Library"="C:\\WINDOWS\\system32\\inetsrv\\w3ctr.s.dll"
"Open"="OpenW3PerformanceData"
```

Appendix C – Tunable Parameters

```
"Close"="CloseW3PerformanceData"
"Collect"="CollectW3PerformanceData"
"PerfIniFile"="w3ctrs.ini"
"Last Counter"=dword:00000a9e
"Last Help"=dword:00000a9f
"First Counter"=dword:000009a8
"First Help"=dword:000009a9
"Object List"="2472 2646"
"Library Validation Code"=hex:00,07,89,ab,22,aa,c3,01,00,5e,00,00,00,00,00,00
"WbemAdapFileSignature"=hex:39,e3,6c,2c,b4,be,59,f5,17,7c,c4,d5,2f,dc,f7,1a
"WbemAdapFileTime"=hex:52,ba,5b,ab,22,aa,c3,01
"WbemAdapFileSize"=dword:00005e00
"WbemAdapStatus"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Security]
"Security"=hex:01,00,14,80,90,00,00,00,9c,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,00,01,01,00,00,00,00,00,01,00,00,\
  00,00,02,00,60,00,04,00,00,00,00,00,14,00,fd,01,02,00,01,01,00,00,00,00,\
  05,12,00,00,00,00,18,00,ff,01,0f,00,01,02,00,00,00,00,00,05,20,00,00,00,\
  20,02,00,00,00,00,14,00,8d,01,02,00,01,01,00,00,00,00,00,05,0b,00,00,00,00,\
  00,18,00,fd,01,02,00,01,02,00,00,00,00,00,05,20,00,00,00,23,02,00,00,01,01,\
  00,00,00,00,00,05,12,00,00,00,01,01,00,00,00,00,00,05,12,00,00,00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC\Enum]
"0"="Root\\LEGACY_W3SVC\\0000"
"Count"=dword:00000001
"NextInstance"=dword:00000001
```

RTE Input Parameters

BenchCraft Configuration File

```
Profile: 1800_6_1_01_2
File Path: C:\benchcrf\1800_6_1_01_2.pro
Version: 4
```

Number of Engines: 6

```
Name: DRIVER1
Description: rte6
Directory: c:\tpcclog\rte6.log
Machine: rte6
Parameter Set: PARAM2
Index: 0
Seed: 59915
Configured Users: 3400
Pipe Name: DRIVER1958504807
Connect Rate: 2000
Start Rate: 1000
Max. Concurrency: -1
Concurrency Rate: 10
CLIENT_NURAND: 233
```

Appendix C – Tunable Parameters

CPU: 0
Additional Options:

Name: DRIVER2
Description: rte2
Directory: c:\tpcclog\rte2.log
Machine: rte2
Parameter Set: PARAM2
Index: 100000000
Seed: 59915
Configured Users: 3400
Pipe Name: DRIVER2958566445
Connect Rate: 2000
Start Rate: 1000
Max. Concurrency: -1
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0
Additional Options:

Name: DRIVER3
Description: rte3
Directory: c:\tpcclog\rte3.log
Machine: rte3
Parameter Set: PARAM2
Index: 200000000
Seed: 59915
Configured Users: 3400
Pipe Name: DRIVER3958590900
Connect Rate: 2000
Start Rate: 1000
Max. Concurrency: -1
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0
Additional Options:

Name: DRIVER4
Description: rte4
Directory: c:\tpcclog\rte4.log
Machine: rte4
Parameter Set: PARAM2
Index: 300000000
Seed: 59915
Configured Users: 3400
Pipe Name: DRIVER41824367832
Connect Rate: 2000
Start Rate: 1000
Max. Concurrency: -1
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0
Additional Options:

Name: DRIVER5

Appendix C – Tunable Parameters

Description: rte5
Directory: c:\tpcclog\rte5.log
Machine: rte5
Parameter Set: PARAM2
Index: 400000000
Seed: 59915
Configured Users: 3400
Pipe Name: DRIVER52001196187
Connect Rate: 2000
Start Rate: 1000
Max. Concurrency: -1
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0
Additional Options:

Name: DRIVER6
Description: rte7
Directory: c:\tpcclog\rte7.log
Machine: rte7
Parameter Set: PARAM2
Index: 500000000
Seed: 59915
Configured Users: 1000
Pipe Name: DRIVER6-339553843
Connect Rate: 2000
Start Rate: 1000
Max. Concurrency: -1
Concurrency Rate: 10
CLIENT_NURAND: 233
CPU: 0
Additional Options:

Number of User groups: 11

Driver Engine: DRIVER1
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1 - 170
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER1
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML

Appendix C – Tunable Parameters

w_id Range: 171 - 340
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER2
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 341 - 510
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER2
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 511 - 680
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER3
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 681 - 850
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER3
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa

Appendix C – Tunable Parameters

Protocol: HTML
w_id Range: 851 - 1020
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER4
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1021 - 1190
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER4
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1191 - 1360
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER5
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1361 - 1530
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER5
IIS Server: client77
SQL Server: pe2650
Database: tpcc

Appendix C – Tunable Parameters

User: sa
Protocol: HTML
w_id Range: 1531 - 1700
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1700
District id: 1
Scale Down: No

Driver Engine: DRIVER6
IIS Server: client77
SQL Server: pe2650
Database: tpcc
User: sa
Protocol: HTML
w_id Range: 1701 - 1800
w_id Min Warehouse: 1
w_id Max Warehouse: 1800
Scale: Normal
User Count: 1000
District id: 1
Scale Down: No

Number of Parameter Sets: 5

~Default

Default Parameter Set

	Txn Weight	Think Time	Key Time	RT Delay	RT Delay	Menu Fence	Menu Delay	
New Order	10.00	10.00	12.05	18.01	0.10	5.00	0.10	
Payment	10.00	10.00	12.05	3.01	0.10	5.00	0.10	
Delivery	1.00	1.00	5.05	2.01	0.10	5.00	0.10	
Stock Level	1.00	1.00	5.05	2.01	0.10	20.00	0.10	
Order Status	1.00	1.00	10.05	2.01	0.10	5.00	0.10	

PARAM2

	Txn Weight	Think Time	Key Time	RT Delay	RT Delay	Menu Fence	Menu Delay	
New Order	44.84	44.84	12.04	18.02	0.10	5.00	0.10	
Payment	43.04	43.04	12.04	3.02	0.10	5.00	0.10	
Delivery	4.05	4.05	5.04	2.02	0.10	5.00	0.10	
Stock Level	4.05	4.05	5.04	2.02	0.10	20.00	0.10	
Order Status	4.05	4.05	10.04	2.02	0.10	5.00	0.10	

Appendix C – Tunable Parameters

Appendix D – Disk Storage

Appendix D – Disk Storage

TPC-C 60 Day Space Requirements						
Warehouses	1900				TpmC	22,052.81
Table	Rows	Data KB	Index KB	Extra 5% KB	8hr Space	Total Space KB
Warehouse	1900	208	24	12		244
District	19000	2112	24	107		2243
Customer	57000000	41454552	2471888	2,196,322		46122762
History	57000000	3166680	16		588,077	3166696
NewOrder	17100000	270360	632			270992
Orders	57000000	1747128	794488		324,455	2541616
OrderLine	570001928	35625128	75424		6,615,867	35700552
Item	100000	9528	40	478		10046
Stock	190000000	60800000	113576	3,045,679		63959255
Total		143,075,696	3,456,112	5,242,598	7,528,400	151,774,406
MB						
Dynamic Space	39,589	Sum of Data for Order, Orderline and History				
Static Space	108,628	Sum of Data+Index+5%-Dynamic Space				
Free Space	na	Total Allocated Spac - (Dynamic + Static Space)				
Daily Growth	7,352	(Dynamic Space/(W*62.5))*tpmc				
Daily Spread	-	(Free Space -1.5*Daily Growth) Zero Assumed				
60 Day Space MB	549,746					
60 Day Space GB	536.86	GB				
Log Size	54,000	MB				
KB Per New Order	4.9674	KB				
8 hr log MB	51,349	MB				
8 hr log GB	50.1453	GB				
Space Usage	GB Needed	Disks Measured	GB Priced	Disk Size	Formatted Size	
60 Day Space DB	536.86	56	944.89	18GB	16.873	
Total DB		56.00	944.89	9GB		
8-hr log + mirror	100.2906	4	134.98	36GB	33.746	
OS, Swap	3	1	8.44	9GB	8.437	
Total Storage	640.15	GB	1,088.31	GB		

Appendix E – Price Quotations

Appendix E - Price Quotations


Appendix E – Price Quotations

Shopping Cart - Microsoft Internet Explorer

Address: <http://order.store.yahoo.com/cgi-bin/wg-order?Unique=ef9428&catalog=lanadapters&et=417e704a8&basket=b%3D5C50808d800ad49417e6941ccdc518088d813e2a68f0410716a5fb4c338c5d6%26%3D%2>

LanAdapters.com

NEW! [Send](#) to more than one address. [What's This?](#)

Item	Options	Unit Price	Quantity	Subtotal	
 7Ft Category 5e cross over RJ45/RJ45 PC To PC Cat5 LIFETIME WARRANTY 3ft available also Crossover Cable with molded ends (backwards compatible with cat5)	Select_color: gray	1.60	<input type="text" value="3"/>	4.80	Remove
Subtotal for LanAdapters.com				4.80	

Appendix E – Price Quotations

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

Tel 425 882 8080
Fax 425 936 7329
<http://www.microsoft.com/>

Microsoft

October 15, 2004

Dell
Kong Yang
1 Dell Way
Round Rock, TX 78682

Mr. Yang:

Here is the information you requested regarding pricing for several Microsoft products to be used in conjunction with your TPC-C benchmark testing.

All pricing shown is in US Dollars (\$).

Part Number	Description	Unit Price	Quantity	Price
228-01079	SQL Server 2000 Standard Edition <i>Per Processing Licensing</i> <i>Discount Schedule: No Discounts Applied</i>	\$4,999	1	\$4,999
P73-00295	Windows Server 2003, Standard Edition <i>Server License Only - No CALs</i> <i>Discount Schedule: Open Program - No Level</i> <i>Unit Price reflects a 26% discount from the</i> <i>retail unit price of \$999.</i>	\$738	2	\$1,476
254-00170	Visual C++ Standard Edition <i>Discount Schedule: No Discounts Applied</i>	\$109	1	\$109
	Microsoft Problem Resolution Services <i>Professional Support</i> <i>(1 incident)</i>	\$245	1	\$245

All products are currently orderable through Microsoft's normal distribution channels.

This quote is valid for the next 90 days.

If we can be of any further assistance, please contact Jamie Reding at (425) 703-0510 or jamiere@microsoft.com.

Reference ID: PCkoya0415100150

Please include this Reference ID in any correspondence regarding this price quote.