July 1998
ECG099/0698

Prepared by Database Technology and Solutions Center

Compaq Computer Corporation

Contents

Supporting Thousands of Sybase ASE Users on Compaq ProLiant Servers

The Compaq Database Technology and Solutions Center (DTSC) has joined forces with Sybase to identify techniques and practices used in supporting thousands of concurrent Sybase ASE users on Compaq ProLiant Servers.

With the continued growth of Microsoft Windows NT based applications and solutions, there is an increasing need to support enterprise class applications on Windows NT systems and servers. The need to support large user communities creates challenges not only for the RDBMS environment, but the entire information technology (IT) infrastructure. This White Paper evaluates the capability of Sybase's Adaptive Server Environment (ASE) in supporting enterprise class applications.

A series of tests were performed to address key issues such as scalability, performance, and system requirements as well as management and administration issues. The tests focused on the following objectives:

- Determine the number of concurrent users that Sybase ASE can support.
- Identify the performance implications for supporting thousands of users.
- Determine the system memory requirements when supporting thousands of concurrent Sybase ASE users.
- Identify system management and database administration issues that should be considered when supporting large user communities.

The test results conclude that Sybase ASE 11.5.1 with it's enhanced features and functionality is capable of supporting large user communities. Key findings are listed below:

- Sybase ASE provides the capacity and functionality necessary to support thousands of concurrent users Windows NT enterprise.
  - Sybase ASE achieved 5,000 concurrent users under load
- Other Sybase ASE enterprise performance and scalability features
  - Enhanced database cache management and proxy login support for middle-tier components.
  - Unified Sybase/NT login (single password for both)
  - Integrated E-mail support for sending messages and receiving incoming queries
  - Integrated Sybase/NT login administration using Extended Stored Procedures

Notice

The information in this publication is subject to change without notice and is provided “AS IS” WITHOUT WARRANTY OF ANY KIND. THE ENTIRE RISK ARISING OUT OF THE USE OF THIS INFORMATION REMAINS WITH RECIPIENT. IN NO EVENT SHALL COMPAQ BE LIABLE FOR ANY DIRECT, CONSEQUENTIAL, INCIDENTAL, SPECIAL, PUNITIVE OR OTHER DAMAGES WHATSOEVER (INCLUDING WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION OR LOSS OF BUSINESS INFORMATION), EVEN IF COMPAQ HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The limited warranties for Compaq products are exclusively set forth in the documentation accompanying such products. Nothing herein should be construed as constituting a further or additional warranty.

This publication does not constitute an endorsement of the product or products that were tested. The configuration or configurations tested or described may or may not be the only available solution. This test is not a determination or product quality or correctness, nor does it ensure compliance with any federal state or local requirements.

Product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

Compaq, Contura, Deskpro, Fastart, Compaq Insight Manager, LTE, PageMarq, Systempro, Systempro/LT, ProLiant, TwinTray, ROMPaq, LicensePaq, QVision, SLT, ProLinea, SmartStart, NetFlex, DirectPlus, QuickFind, RemotePaq, BackPaq, TechPaq, SpeedPaq, QuickBack, PaqFax, Presario, SilentCool, CompaqCare (design), Aero, SmartStation, MiniStation, and PaqRap, registered United States Patent and Trademark Office.

Netelligent, Armada, Cruiser, Concerto, QuickChoice, ProSignia, Systempro/XL, Net1, LTE Elite, Vocalyst, PageMate, SoftPaq, FirstPaq, SolutionPaq, EasyPoint, EZ Help, MaxLight, MultiLock, QuickBlank, QuickLock, UltraView, Innovate logo, Wonder Tools logo in black/white and color, and Compaq PC Card Solution logo are trademarks and/or service marks of Compaq Computer Corporation.

Microsoft, Windows, Windows NT, Windows NT Server and Workstation, Microsoft SQL Server for Windows NT are trademarks and/or registered trademarks of Microsoft Corporation.

NetWare and Novell are registered trademarks and intraNetWare, NDS, and Novell Directory Services are trademarks of Novell, Inc.

Pentium is a registered trademark of Intel Corporation.

Supporting Thousands of Sybase ASE Users on Compaq ProLiant Servers
White Paper prepared by:

Sybase Corporation

Database Technology and Solutions Center

AUTHOR: Edward Bueche (Sybase Corporation)

First Edition (July 1998)
Document Number ECG099/0698

INTRODUCTION

This document outlines basic issues, techniques, and considerations for supporting thousands of users with Sybase ASE 11.5.1 for Windows NT.

Thousands of concurrent users challenge both the performance and the ability to manage 2-tier and 3-tier systems. Each additional user connection consumes more memory and processor capacity. Also, provisioning and supporting thousands of users can turn simple administrative tasks into tedious, error prone chores. System management problems begin with client setup where geographically dispersed machines can have different operating systems and different application versions, and continue to the database server where not only scalability, but also security, issues must be addressed.

The Adaptive Server Enterprise 11.5.1 provides the capacity and functionality necessary to support such installations in the Windows NT Enterprise. These features include support for:

- Thousands of simultaneous connected users
- Up to 3GB memory for data cache; configurable procedure cache and task stacks
- Integrated E-mail which can send and receive E-mail
- Unified Sybase/NT login (single password for both)
- Integrated Sybase/NT login administration using Extended Stored Procedures
- Proxy authorization for middle tier servers

The Enterprise support of Sybase Adaptive Server Enterprise for Windows NT is further enhanced with the new parallel features 11.5 (Parallel Query, sort, index creation, and load), mixed workload support (Engine Groups, Resource Governors), and the Active/Active Cluster support for high availability systems.

BACKGROUND

Because of the current trend toward an increasing number of users, system architects must employ robust 2-tier or 3-tier systems as appropriate. A 2-tier client/server system is shown below. The user community logs directly into the DBMS server. In this case, ASE manages all of the user connections and their resources. The primary benefit of this architecture is simplicity: there are fewer server software components to administer and troubleshoot. ASE supports thousands of simultaneous connections in this type of environment. More details on an example of this setup will be covered later.

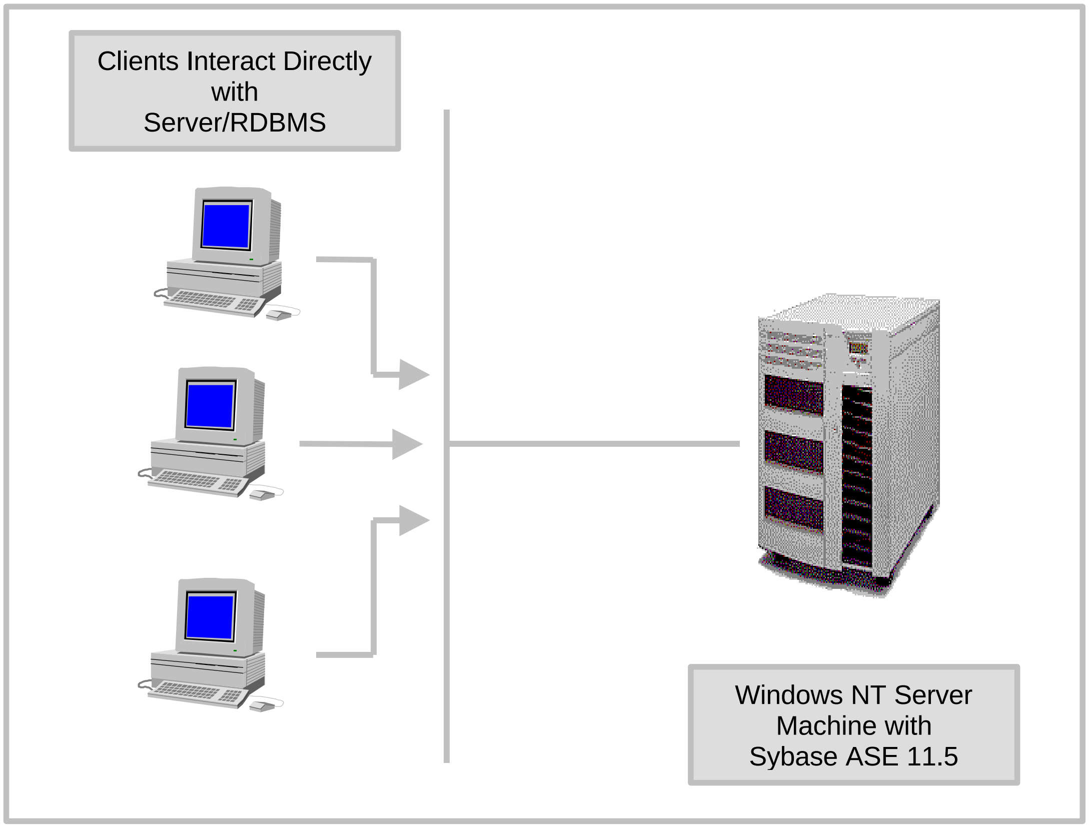

Clients Interact Directly with Server/RDBMS
Windows NT Server Machine with Sybase ASE 11.5

Figure 1 – Typical 2-Tier Client/Server Architecture

There are two downsides of this architecture in environments with thousands of users. First, ensuring that all client machines have the proper application and DBMS software can be challenging. Users can be geographically dispersed with different versions of an operating system (Windows 3.1, Windows 95, Windows NT, Mac OS, UNIX). Second, although not all users will be active at one time, they are likely to be consuming extra memory on the server even if their connection is idle.

Three-tier systems offer several features that address some of the disadvantages of 2-tier client/server systems in large-scale deployments. An example of a 3-tier client/server system is shown below. Clients log into a middle tier server (web server, transaction manager, or application specific server) and this server logs in to the DBMS server on behalf of the clients. One advantage of this architecture is simplified client administration (in many cases the only software the client needs is a web browser the rest is downloaded from the middle tier server). This is a recent development in middle tier technology (and is a chief advantage of the Sybase Jaguar Component Transaction Server). The main function of many older middle tier servers was to contain the business logic of the application in a central server.

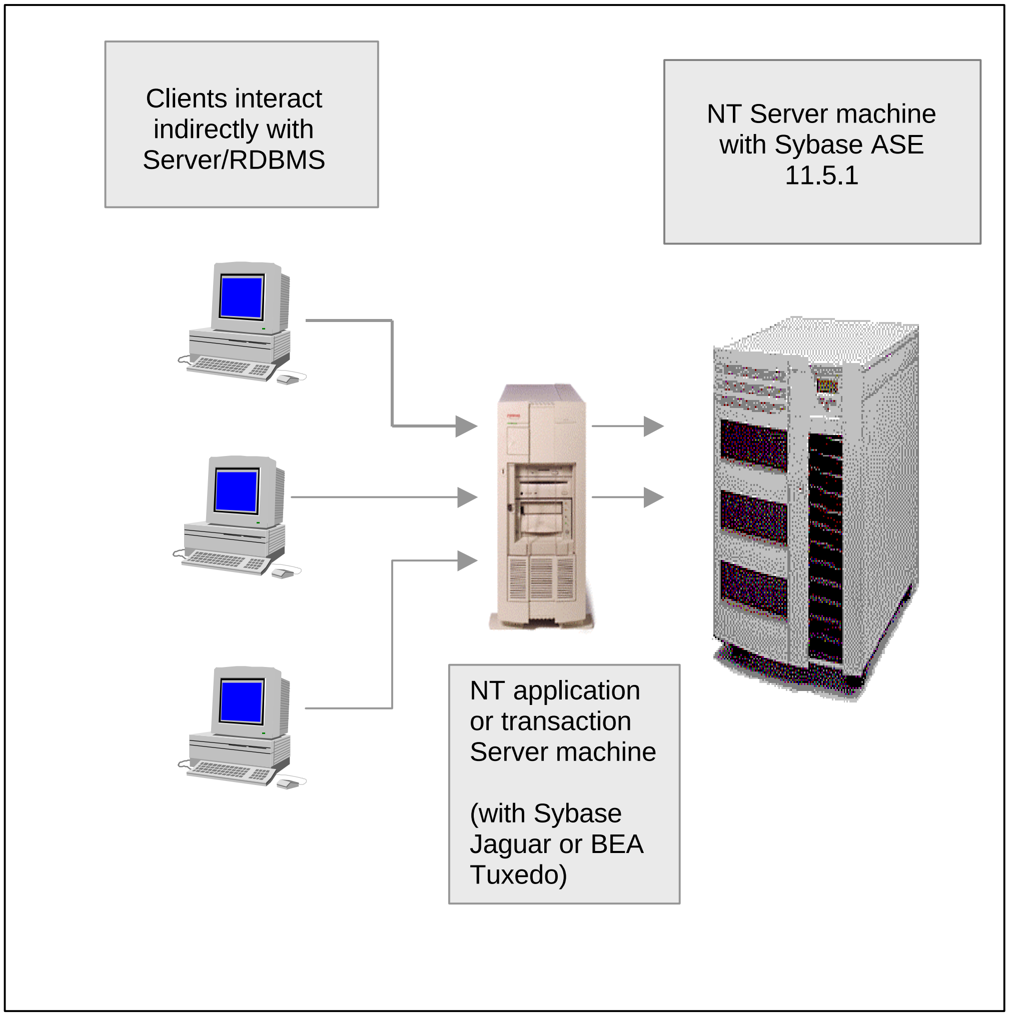

Clients interact indirectly with Server/RDBMS
NT Server machine with Sybase ASE 11.5.1
NT application or transaction Server machine
(with Sybase Jaguar or BEA Tuxedo)

Figure 2 – 3-Tier Architecture

Another advantage is the ability to share (or multiplex) many user connections into fewer database connections. That is, the middle tier server can maintain a smaller pool of permanent connections to the database. It then directs any incoming client requests down one of the permanent connections. This avoids any new connection setup and allows the DBMS to have fewer connections therefore reducing the DBMS memory requirements. This typically works well because in systems with thousands of concurrent connections, most connections are typically idle at any one time. For example, the TPC-C benchmark run with ASE 11.5 on Windows NT supported 6,400 users through BEA's Tuxedo middle tier server multiplexed to less than 200 Sybase connections[1]

There are, of course, some security issues when sharing a DBMS connection among different users. The proxy authorization feature in ASE allows the shared DBMS connection to change its access and security rights depending on which user is being served. Hence, multiple users across different applications can effectively and safely share the common pool of DBMS connections in the middle tier server.

However, this sharing (or multiplexing) of connections is by no means an essential component of 3-tier systems. A middle tier server could create one or more DBMS connections for each user connection. Hence, a large number of users could result from either 2-tier or 3-tier architectures.

In a mobile replication environment for Adaptive Server Enterprise, a large user community could arise. SQL Remote is the name of the mobile, "occasionally connected" data replication mechanism offered by ASE. A single ASE site might support thousands users of a mobile workforce via SQL Remote. In the following example, since the users are not connected all the time, it is unlikely that all users would log in at once. Hence, the number of concurrent users in the database is going to be smaller than the total number of actual users defined in the database. This architecture is optimal for environments with a mobile workforce, such as a direct sales force, and for environments where branch offices with minimal hardware setup only need to connect occasionally to the central office.

[1] This benchmark published March 1997 listed Sybase SQL Server 11.1 which was the name of Adaptive Server Enterprise 11.5 before it was renamed. Throughput 7,650 TPM-C, $82 per TPM-C, Availability 9/97. TPC-C is a trademark of the Transaction Processing Council.

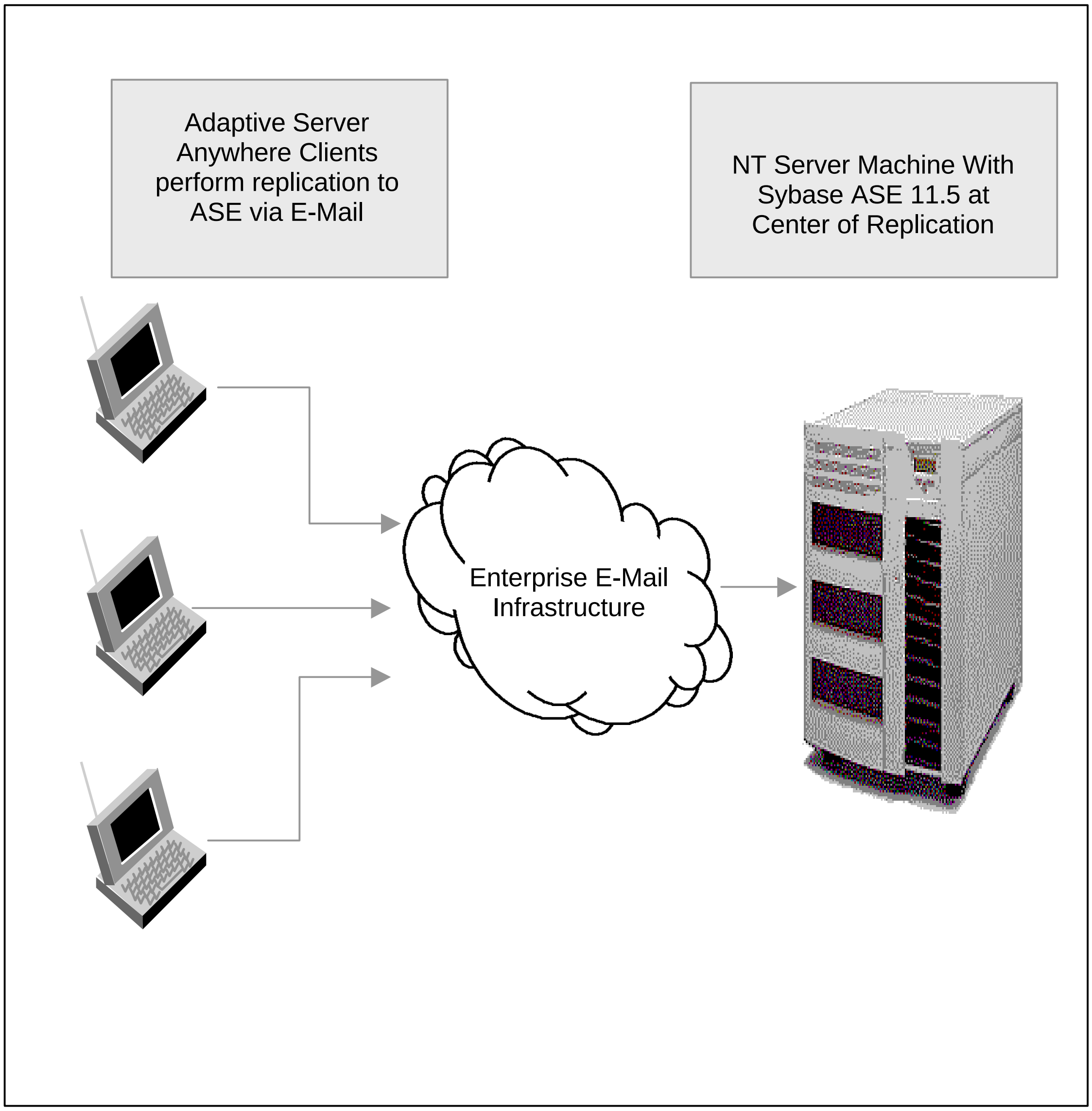

Adaptive Server Anywhere Clients perform replication to ASE via E-Mail
NT Server Machine With Sybase ASE 11.5 at Center of Replication
Enterprise E-Mail Infrastructure

Figure 3 – ASE Replication with E-Mail Clients

Even though the number of simultaneous users may not be large in this case, several administrative issues are common to ones that show up in any large user community. For example, suppose that a planned outage occurs, such as a software upgrade of a business application. The DBA needs to notify every user of the database. Sybase Adaptive Server Enterprise is integrated with E-mail, so the DBA can easily notify the database user community of any pending maintenance. This feature is also extremely useful in serving Internet applications that want to contact potential customers or newly registered customers via E-mail.

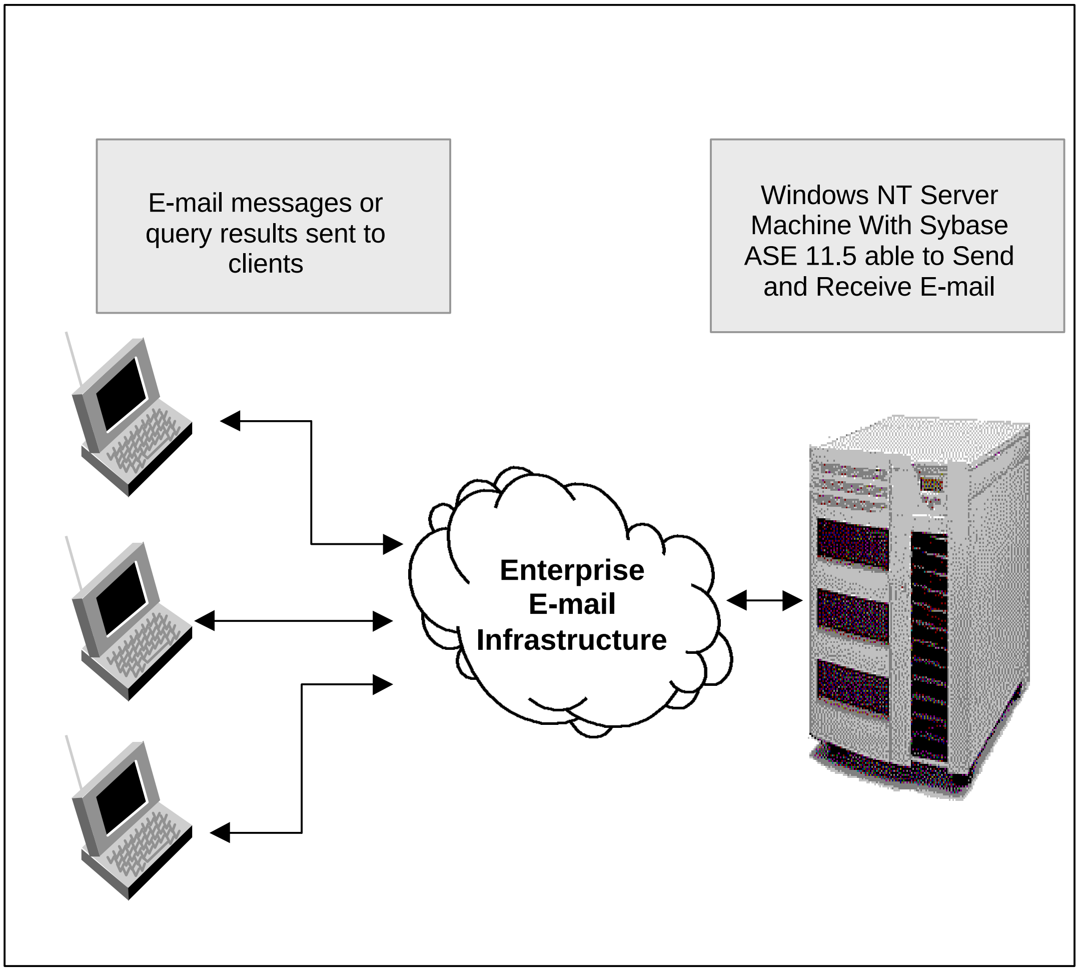

E-mail messages or query results sent to clients
Windows NT Server Machine With Sybase ASE 11.5 able to Send and Receive E-mail
Enterprise E-mail Infrastructure

Figure 4 – Remote User Database Access

Furthermore, since Sybase ASE can receive e-mail queries and send results via e-mail, a large heterogeneous community of users can be served using just the existing corporate e-mail infrastructure. Although response time is probably the longest of all configurations discussed above, client setup overhead is very low. Hence, this architecture can be extremely useful for applications with large user communities, with mobile users, with users that are difficult to administer, and with users with relaxed response time requirements.

System availability becomes a significant issue when serving a large user community. By definition any hardware or software outage could affect thousands of users. Sybase Adaptive Server Enterprise for Windows NT supports an Active/Active configuration using Microsoft Cluster Server (covered in more detail in the Sybase white paper: *Configuring Sybase Adaptive Server Enterprise for High Availability: Microsoft Cluster Server*) and has several features to minimize database recovery time. The fast recovery features include using asynchronous pre-fetch during recovery and free checkpoints using the housekeeper internal DBMS task.

The more concurrent users supported on a single NT system, the more likely that these users will be performing very different tasks. Some may be Online Transaction Processing (OLTP) users with strict response time requirements, while others will be executing Decision Support (DSS) queries against the database. Parallel query allows DSS users to consume far more DBMS resources than OLTP users.

Adaptive Server Enterprise ensures excellent DSS and OLTP performance with its mixed workload features that include: the Logical Memory Manager, the Logical Process Manager, and Resource Governors. A Logical Memory Manager (LMM) lets you assign database objects to an unlimited number of named caches that you configure. High priority data remains in cache for improved response times. The Logical Process Manager (LPM) allows you to assign CPU resources to individual applications so low priority work does not impact high priority work, ensuring more predictable performance (also known as Engine Groups). Resource Governors prevent run away queries that can tie up resources needed for rapid response rates. Managing resource consumption for queries, batches, or transactions allows you to specify limits based on I/O cost, elapsed time, or number of rows returned.

CONFIGURATION ISSUES & USEFUL FEATURES

The following section describes the configuration issues for supporting a large number of users.

Useful Documentation

The following documents are recommended reading & reference material for information system professionals setting up large user applications using Sybase ASE on Windows NT:

- Sybase Adaptive Server Enterprise Performance and Tuning Guide
  - ❑ Sybase Document ID# 32645-01-1150-02
- Configuring Sybase Adaptive Server Enterprise for Windows NT
  - ❑ Sybase Document ID# 32645-01-1150 Part Number AA0539
- Sybase Adaptive Server(TM) Enterprise System Administration Guide
  - ❑ Sybase Document ID# 32645-01-1150
- Sybase Adaptive Server(TM) Enterprise Security Administration Guide
  - ❑ Sybase Document ID# 32645-01-1150
- Windows NT Server Resource Kit

Useful Tools

We recommend that you familiarize yourself with the following tools:

- **sp_sysmon**: displays internal performance information on the dataserver as a workload is executed. The counters are normalized by second and by transaction. For details, see the *Sybase Adaptive Server Enterprise Performance and Tuning Guide*.

- **sp_configure**: displays the current memory usage for some internal structures and caches. For details, see the *Sybase Adaptive Server Enterprise Reference Manual.*

- **dbcc showcache**: can be used to indicate if any data cache memory has not been referenced..

User Connections

The number of user connections to the database is set using the sp_configure T-SQL command:

sp_configure “user connections”, <value>

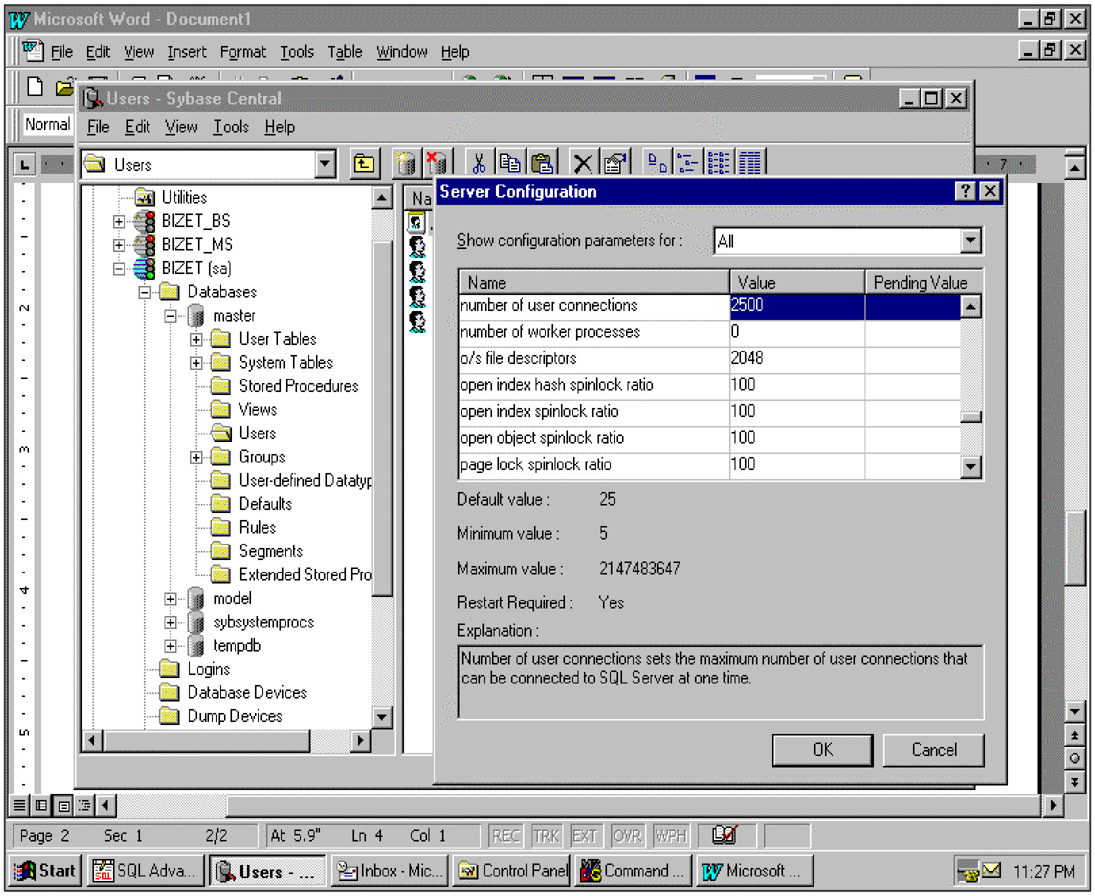
For example:

```
sp_configure “user connections”, 2500
```

The server needs to be rebooted in order to allow this change to take place.

Each user connection requires about 45K of address space in dataserver internal structures, a default of 128K of DBMS task stack (allocated by NT), and perhaps some additional space for query plans, stored procedure plans, and dynamic SQL (this additional space is workload dependent). The DBMS task stack is not allocated until the connection is created.

To set the **user connections** parameter from the SQL Central tool, right click on the server object and select the **configure** option in the popup menu.



Server Configuration

Show configuration parameters for : All

Name	Value	Pending Value
number of user connections	2500	
number of worker processes	0	
o/s file descriptors	2048	
open index hash spinlock ratio	100	
open index spinlock ratio	100	
open object spinlock ratio	100	
page lock spinlock ratio	100	

Default value : 25

Minimum value : 5

Maximum value : 2147483647

Restart Required : Yes

Explanation :

Number of user connections sets the maximum number of user connections that can be connected to SQL Server at one time.

1. *What is the maximum value I can set for* **user connections***?*

   Currently on ASE 11.5.1 it is 12,000. However, this does not imply that ASE for Windows NT is able to support 12,000 users from your application, due to various other resource constraints. In this paper we cover some performance data from runs with up to 5,000 users. By setting the number of users up higher, you may have to adjust other memory parameters to accommodate the extra memory used by the larger number of users.

2. *Is there any penalty for over configuring the number of users?*

   If the **user connections** parameter is set to a much higher value than needed, a large amount of memory may be wasted. This wasted memory is then not available for such needs as the procedure cache or the data caches. Each user connection consumes at least 45KB in static structures, even if the connection is not established. For example, if the user community of concurrent users is no more than 2,500 and **user connections** is set to 5,000, then about 112 MB of space is wasted.

   One rule of thumb is to configure about 5 – 10% more connections than the expected maximum number of user connections.

3. *How do I know if the number of user connections was set too small?*

   If new clients are unable to log into the server, that is their connection is refused once the maximum number of clients has logged in, then the number of user connections is set too small.

Memory Issues

This section covers memory issues related to large numbers of users. As mentioned earlier, more concurrent users consume more memory on the DBMS server. The following figure shows how Adaptive Server for Windows NT uses memory. On NT Adaptive Server is a single process address space shared by a number of engine NT threads and some communication NT threads. Each user connection is an NT fiber, and these NT fibers can run on any of the DBMS engines.

The **total memory** configuration parameter specifies the total amount of memory consumed by the dataserver for data caches, procedure cache, program text, and internal. For Adaptive Server on NT the **total memory** parameter does not include the memory for DBMS task (or NT fiber) program stacks, as it does for UNIX versions of this product, because each fiber is assigned a program stack by NT. For like amounts of physical memory, the total memory configured for a Sybase ASE NT server needs to be less than that for an ASE for UNIX server. Since Windows NT assigns the task stacks, the only way to control their size is to use the NT EDITBIN utility.

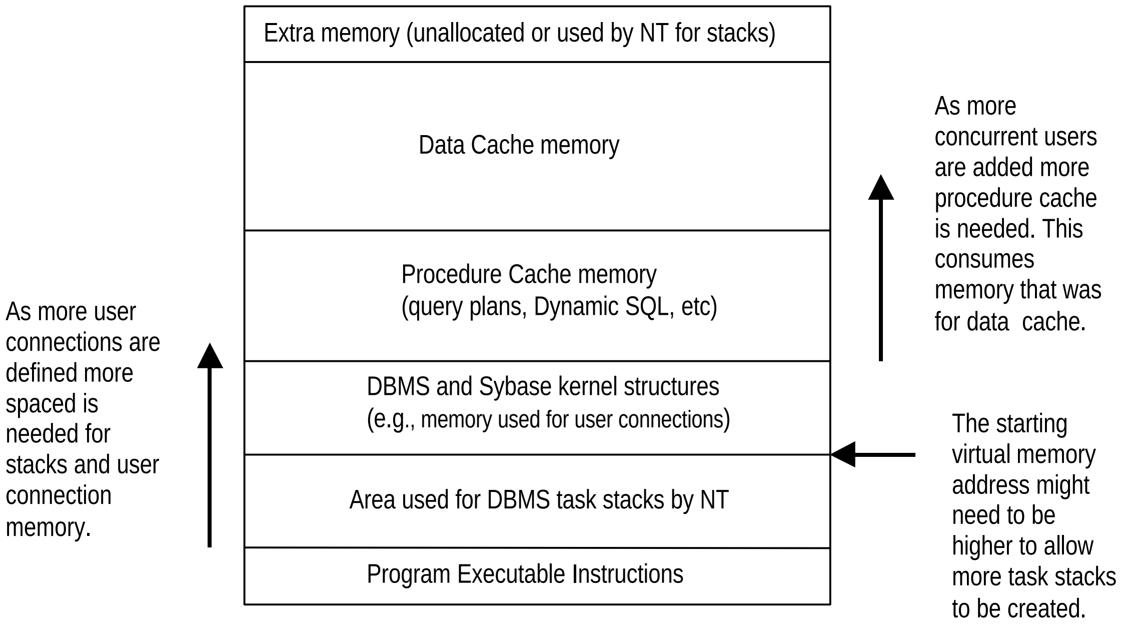

Extra memory (unallocated or used by NT for stacks)
Data Cache memory
Procedure Cache memory
(query plans, Dynamic SQL, etc)
DBMS and Sybase kernel structures
(e.g., memory used for user connections)
Area used for DBMS task stacks by NT
Program Executable Instructions
As more user connections are defined more spaced is needed for stacks and user connection memory.
As more concurrent users are added more procedure cache is needed. This consumes memory that was for data cache.
The starting virtual memory address might need to be higher to allow more task stacks to be created.

Figure 5 - Address Space for Sybase Adaptive Server Enterprise for Windows NT

A certain amount of control is provided over these various pieces of the address space. Because this memory is statically allocated upon startup of Adaptive Server, any changes you make to these parameters require restarting the server. As you add more concurrent users more memory is needed for DBMS task stacks, for Sybase internal data structures, and for procedure cache. The more memory used for these, the less that can be used for the data cache.

Each memory use may need to be tuned separately taking into account the different user workloads. In addition to raising the value of the **user connections** parameter, you may also have to increase the percentage of procedure cache and modify the size of the DBMS task stacks.

Section of Memory	**Configure Size? How?**	**Configure Placement ? How?**	**Notes**
Total memory	Yes **sp_configure** or Sybase Central	Yes The **starting virtual memory address** parameter in **sp_configure** indicates where this section of memory starts.	On NT Server Enterprise 4.0 the Adaptive Server supports the 3 GB user address space. Restart Adaptive Server when **virtual memory address** changes.
Program text	No	No	
DBMS task stack	Yes EDITBIN /Stack:<size> sqlsrvr.exe	No NT picks where these are placed.	On UNIX the stack size is set via **sp_configure**. In both UNIX and NT restart the server after changing the stack size.
ASE kernel and DBMS structures	Yes **sp_configure** or Sybase Central	Yes The **starting virtual memory address** parameter in **sp_configure** indicates where this section of memory starts.	The default **starting virtual memory address** is 500MB. Restart Adaptive Server when **virtual memory address** changes.
Procedure cache	Yes **sp_configure "procedure cache percent"**	No	Specify the percentage of total memory that is dedicated to the procedure cache. Restart Adaptive Server when **procedure cache percent** changes**.**
Data cache	Yes **sp_configure, sp_cacheconfig**	No	Data caches are always allocated after the procedure cache and other structures.

Table 1 ASE Memory Configuration Parameters

Frequently asked questions about Memory:

How do I know if **total memory** *is set too high?*

If **total memory** is set too high for Adaptive Server Enterprise on Windows NT the following problems could occur:

- OS paging could result. The size of the ASE address space is the sum of **total memory** and the space consumed by NT fiber stacks. The memory on the server machine is going to be consumed partially by the OS and any other applications that run on the system. If there is not sufficient memory on the system to accommodate all applications, the OS, and ASE, then the OS starts paging memory out to the paging file. You can see this behavior using the NT Performance Monitor.

  Solutions to OS paging include adding more physical memory, not running some of the other applications, or reducing the **total memory** setting in Adaptive Server Enterprise.

- ASE might have trouble spawning some connections. This happens if the amount of **total memory** allocated for ASE does not leave sufficient address space for the NT task stacks.

- ASE does not boot because it cannot obtain sufficient virtual memory. Sometimes the cause is not too large a **total memory** setting; rather, it can be due to the default setting of the **starting virtual memory address** parameter (500MB). This parameter controls where in the address space ASE starts to allocate memory. If it is set too high, then ASE might not be able to find sufficient space in the rest of the address space.

How do I know if **total memory** *is set too low?*

If the **total memory** parameter is set too low then the server might not start, because memory is insufficient for allocating some cache or internal data structure.

How much memory should I configure for my data caches?

This is clearly quite application specific. For an excellent set of guidelines, see the details on memory use and performance in the *Sybase Adaptive Server Enterprise Performance and Tuning Guide*.

How much procedure cache should I configure?

Configure an amount sufficient to ensure that all users can be serviced. Again, the *Sybase Adaptive Server Enterprise Performance and Tuning Guide* provides an excellent set of guidelines. For best performance ensure sufficient cache, so that pages of a user's plans are not thrown out of the cache to make room for others. You can determine potential short falls in procedure cache from **sp_sysmon** with the following counters.

```
Procedure Cache Management        per sec      per xact       count  % of total
---------------------------  ------------  ------------  ----------  ----------
  Procedure Requests               2540.2          19.5      152411       n/a
  Procedure Reads from Disk           0.0           0.0           0       0.0 %
  Procedure Writes to Disk            0.0           0.0           0       0.0 %
  Procedure Removals                 14.0           0.1         838       n/a
```

If the percentage of procedure reads from disk is large, then the procedure cache could be too small. If the procedure cache is too undersized, the dataserver will generate an error 701.

How can I tell if I don't have sufficient virtual address space for DBMS Task stacks?

This problem could arise if **total memory** is set too high given the actual number of concurrent users. Windows NT creates the DBMS task stacks when DBMS sessions are started, not when the server is started. Two errors are likely to occur:

- Sessions fail as they are being started, and the failure is logged in the error log. For example:

```
00:00000:00014:1998/01/20 14:04:28.60 kernel  Cannot send, host process
disconnected: PERDITA 2858 spid: 14
00:00000:00014:1998/01/20 14:04:28.60 kernel  Cannot send, host process
disconnected: PERDITA 2858 spid: 37
02:00000:00037:1998/01/20 14:04:28.62 server  Error: 1608, Severity: 18,
State: 4
02:00000:00037:1998/01/20 14:04:28.62 server  Error: 1608, Severity: 18,
State: 4
02:00000:00037:1998/01/20 14:04:28.70 server  A client process exited
abnormally, or a network error was encountered. Unless other errors
occurred, continue processing normally.
00:00000:00014:1998/01/20 14:04:28.73 server  A client process exited
abnormally, or a network error was encountered. Unless other errors
occurred, continue processing normally.
```

- If the memory shortfall is great enough, then the dataserver is unable to create any fiber stacks, which appears in the ASE error log. For example:

```
00:00000:00002:1998/01/20 14:30:31.18 kernel  NT operating system error 8
in module 'e:/main/porttree/sql/nt386/ksource/proc/prcontext.c' at line
277: Not enough storage is available to process this command.
```

It is important to note that stacks are also used for the engine threads (a few) and communication threads (could be many).

What can I do If I run out of virtual address space for DBMS Task stacks?

If you have run out of address space for the DBMS task stacks then the following options are available:

- Reduce the DBMS stack size. The default is 128 KB. For some applications you may be able to reduce stack size to 64 KB or 32 KB. You can use the EDITBIN utility to reduce stack size.
- Increase the starting virtual memory address to a higher address value.
- Decrease the amount of memory used for the data caches. Decreasing **total memory** automatically decreases the size of the default data cache. If named caches are used then it may be necessary to decrease their size. You can use SQL Central to decrease named cache size.

How can I tell if I've made my DBMS task stack too small? Too big?

If the DBMS task stack has been set too small then a stack overflow is likely to result during the course of executing the customer workload. ASE traps such events, prints out diagnostic information, and then terminates the connection.

How do I tell how much I have configured for data cache? For procedure cache? For kernel and DBMS structures?

Run **sp_configure**, specifying the “Memory use” group. The output includes:

```
Parameter Name                 Default     Memory Used Config Value Run Value
--------------                 -------     ----------- ------------ ---------
additional network memory                0           0           0            0
lock shared memory                       0           0           0            0
memory alignment boundary             2048           0        2048         2048
memory per worker process             1024           0        1024         1024
shared memory starting address           0           0           0            0
total memory                          9000       36000       18000        18000
number of user connections           10100      423796       10100       10100
procedure cache percent                 10       …………………………………………….
```

Alternatively, you can use the **dbcc memusage** command.

The output includes:

```
Memory Usage:

                         Meg.         2K Blks            Bytes


      Configured Memory:1935.1563      990800      2029158400


              Code size:  1.0000         512       1048576
      Kernel Structures:  0.6065         311        635938
      Server Structures: 15.1721        7769      15909124
           Cache Memory:1898.4746      972019      1990694912
           Proc Buffers:  0.6742         346        706968
           Proc Headers: 19.2266        9844      20160512
```

How can I tell if there is some data cache memory not used?

To tell if some allocated data cache is not being referenced, use the **dbcc showcache** command.

Ensuring that memory is not wasted is even more important when using named caches. If named caches are not used, then all objects use the default data cache. For named caches the DBA binds certain database objects (tables or indexes) to the cache. Once a set of objects are bound to the cache, the cache could be either too small, where all of the buffers are used up during the course of the workload, or larger than necessary. We recommend that the caches for the tables be oversized, if possible, by 15 - 20% to cover any unexpected growth that occurs as application access increases.

Now suppose we want to see how much memory is unused in the default data cache. Execute:

```
dbcc showcache('default data cache')
go
```

The output could be as follows:

```
Cache information for named cache 'default data cache' id 0 using
strict LRU replacement

POOL     IN MRU  IN WASH    KEPT     HASHED  TW      REDIRTY
----------------------------------------------------------
2KB         13167   25687     164       37315   193016  11836
```

The **showcache** command has many diagnostic functions, but our focus here is on its ability to show if caches are over or under sized. The first column, POOL, lists the size of a buffer in the pool, which can be sized at 2 KB, 4 KB, 8 KB, and 16 KB. The fifth column, HASHED, indicates how many buffers have been used. The second, third, and fourth columns give some indication on the state of all buffers. You can compute the unused buffers using the following formula:

$$\textit{Unused Buffers} = (\textit{IN\_MRU} + \textit{IN\_WASH} + \textit{KEPT}) - \textit{HASHED}$$

Using this formula with our output example, we see that the default data cache has 1,703 spare buffers. A common practice is to bind the log to its own cache. The log typically hashes all of its buffers. Actually its working set of buffers is small. The log does not reuse any buffers until it has depleted all available unused buffers.

> ***Note***: Sybase recommends that **showcache** only be run on a system with little or no activity; that is, after the workload has completed. Be sure to review documentation to check the syntax or semantics of **showcache**, which can change from release to release.

APPLICATION SPECIFIC ISSUES

The following are some additional configuration parameters that may need to be tuned as more users are added to the system:

Sp_configure option	Default Value	Increase by	Can determine it needs to be increased when
Locks	5000	More than 2 times the number of users (heavily depends on load)	Error 1204 occurs
Open Objects	500	Hundreds or even thousands at a time.	Sp_sysmon indicates that there are "object reuse requests" going on.

Table 2 – Additional ASE Configuration Parameters

Network Resources

The dataserver ensures sufficient space for each DBMS task for its networking buffers, based on the network parameters **default network packet size** and **max network packet size**, set using **sp_configure**.

1. *How do I know if my packet sizes are large enough?*

   If the packet size used by the client is larger than the **max network packet size** configured on the server, then the client is unable to connect. A message is sent back to the client stating that the packet size it used was too small. Suppose a client tried to login with a packet size of 16KB when the maximum packet size was 4096. A message is returned to the client, such as:

```
The packet size (16384) specified at login is illegal. Legal values are between 512
and 4096.
Msg 4002, Level 14, State 1:
Line 1:
Login failed.
```

2. *If the* **default network packet size** *is less than the* **max network packet size**, *then how do I know if the server is running out of network buffer space?*

   We recommend that you configure the default and maximum packet sizes the same to ensure sufficient space for all network connections when the server starts. If these two parameters cannot be the same size (that is, default packet size > maximum packet size), then the server may run out of network buffers. Without enough network buffers, connections drop as they are being spawned. This condition can arise in the an Enterprise environment if a group of users needs a large packet size for their application, but giving all users buffers that large would waste too much memory.

   To fix this problem, configure **additional network memory**, using **sp_configure**, to allocate additional memory for network buffers. To calculate the size of the additional network memory, use this formula:

   *Additional_netmem = number_of_users_with_large_packets * 3 * maximum_desired_message_size *1.02*

   Using this amount of memory reserves some extra network buffer space for those users with large packets

Logins and Database Users

When you deal with large numbers of users you may need to address setup and security. Most installations set up a separate DBMS login for each user, which can be an administrative hassle. This section covers some hints on how to reduce that hassle.

The security features of the Sybase Adaptive Server Enterprise are covered in more detail in the following two documents:

- *Configuring Sybase Adaptive Server Enterprise for Windows NT* and
- *Sybase(R) Adaptive Server(TM) Enterprise Security Administration Guide*

We will touch upon two that could be important for installations with large numbers of users: integrated security and proxy authorization.

Two main tools help reduce the hassle of setup for a large number of users: T-SQL stored procedures and extended stored procedures. Extended stored procedures, a new feature as of ASE 11.5, allow the server to execute NT programs external to Adaptive Server. If the number of users is small, then the easiest way of adding the users is to employ Sybase Central. However, when adding hundreds or even thousands of users, this approach could become tedious. Instead, set up NT CMD batch or MKS shell scripts to run T-SQL commands to add the users. The scripts run a stored procedure that calls the internal Sybase login/user administration commands then calls external programs to setup the NT logins. The following is an example of a simple stored procedure:

```
        create procedure add_nt_sybase_user
          @name varchar(15),
          @password varchar(15) as

         declare @cmd varchar(80)

# First add user within Sybase
         sp_addlogin @name, @password, mydb
         sp_adduser @name, @name, mydb

         # Second, add user to NT using the Extended stored procedure
         # xp_cmdshell and the NT resource kit utility addusers.
         # The first thing that will happen is that a file is being
         # constructed that will be fed in to addusers

         execute xp_cmdshell "echo [User] > j:\temp\u1.out"
         select @cmd = "echo " + @name + ",," + @password + ",,,, >>
        j:\temp\u1.out"
         execute xp_cmdshell @cmd

         execute xp_cmdshell "echo [Local] >> j:\temp\u1.out"
         select @cmd = "echo " + "Users,,BIZET\" + @name + ">> j:\temp\u1.out"
         execute xp_cmdshell @cmd

         execute xp_cmdshell "addusers /c j:\temp\u1.out"
```

This script first creates the Sybase login and database user using **sp_addlogin** and **sp_adduser**. Then the script uses **xp_cmdshell** to create a load file for the NT resource kit program ***addusers*** and executes ***addusers*** with the file it just created. See the NT resource kit for more details on ***addusers***. You can invoke the above in the following way:

```
execute add_nt_sybase_user    “joe”, “joe1234”
```

Many NT-only sites are likely to be interested in using the integrated security features of ASE and Windows NT. These features allow a single password for the user on NT and Sybase. Of the two ways a single password/user is supported in ASE, the recommended one works over any supported protocol as described in Appendix E: Using Security Services with Windows NT LAN Manager of *Configuring Sybase Adaptive Server Enterprise for Windows NT.*

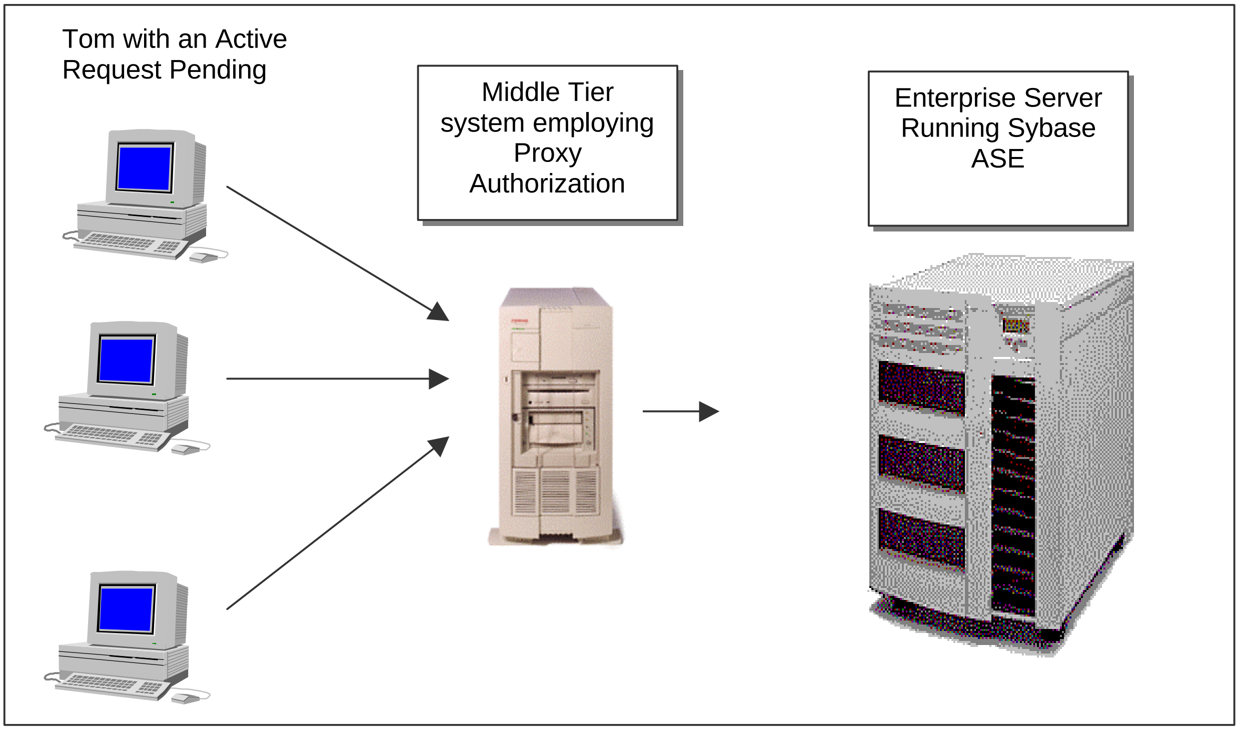
The proxy authorization security feature is useful for 3-tier applications. This feature allows the security personality of an authorized DBMS connection to change depending on which user a middle tier server is servicing. This authorized login can essentially impersonate any other login on Adaptive Server. The example below shows three users: Tom, Mary, and Mark. They access three different applications and databases within Adaptive Server Enterprise and have different access privileges. The middle tier server establishes a connection with Adaptive Server using a login with proxy privileges. Suppose the middle tier server switches from servicing Mary to Tom, by sending the following command on the session:

```
set proxy ‘Tom’
```

Or

```
set session authorization ‘Tom’
```

Once this command executes, the security personality of the session changes from Mary’s to Tom’s.


Tom with an Active Request Pending
Middle Tier system employing Proxy Authorization
Enterprise Server Running Sybase ASE

Integrated E-Mail Helps Support Large User Communities

The Integrated E-mail feature of Adaptive Server Enterprise has many potential useful applications in helping support large user communities. For example, contacting a large number of DBMS users can be a problem. With the Integrated Mail setup, it becomes much simpler for the DBA to contact each user for issues like planned upgrades. This feature is also useful for Internet applications that automatically send E-mail to customers and prospective clients. Also, a user community with varied types of operating systems (such as Windows CE, DOS, Netware, UNIX, NT) and PC types can all send queries and receive results through the existing corporate infrastructure.

For details about the Integrated E-mail feature, see the *Configuring Sybase Adaptive Server Enterprise for Windows NT* guide.

The following sample T-SQL script shows how a DBA can tell all users of a particular database that the system will come down for a planned software upgrade.

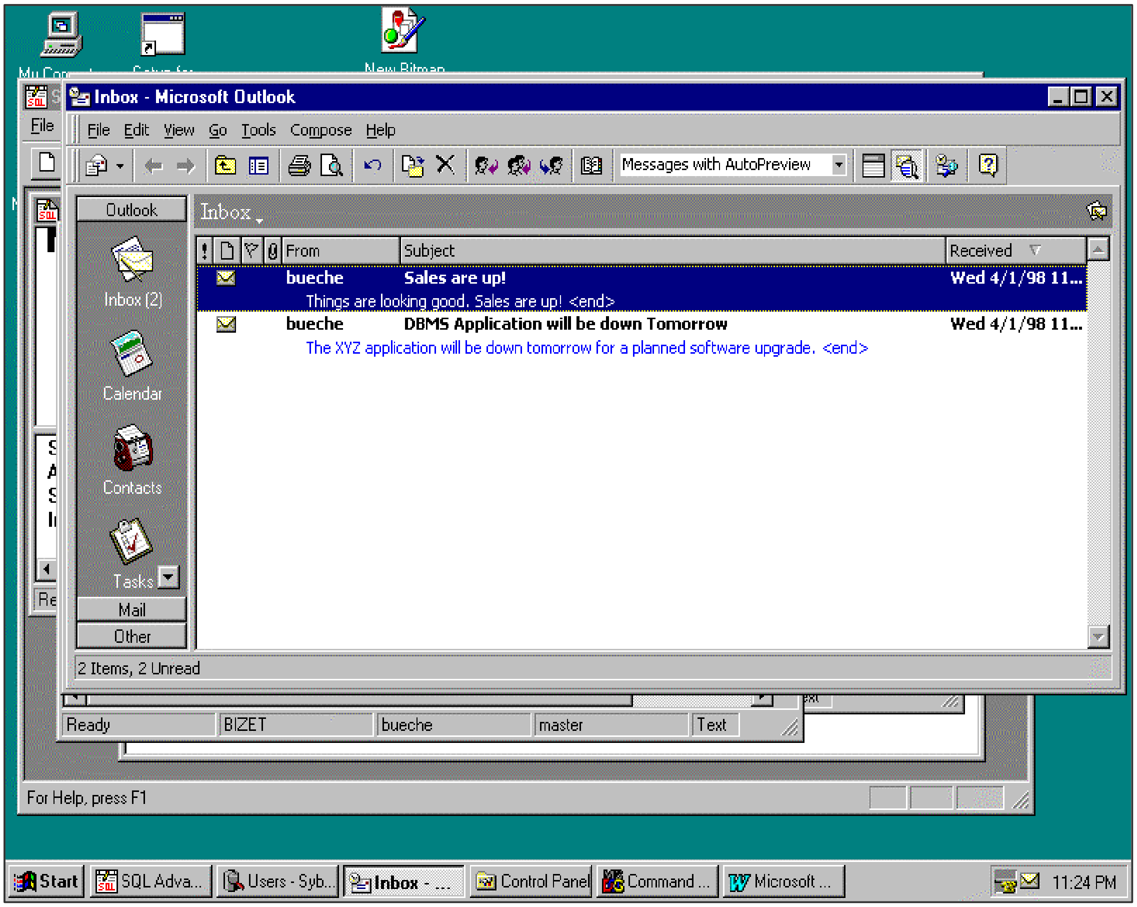
```
declare user_cur cursor
for select name  from  sysusers where suid > 0
go
declare @name char(10)

open user_cur
fetch user_cur into @name
if (@@sqlstatus = 2)
  begin
       print "no users!"
       close user_cur
  end
else
  begin
       while (@@sqlstatus = 0)
       begin
             execute xp_sendmail @subject="downtime",
             @recipient=@name,
             @message="The XYZ application will be down tomorrow for a planned
       software upgrade"

             fetch user_cur into @name
       end
 end
close  user_cur
go
deallocate cursor user_cur
go

 print "All messages sent!"

---------------- Sample Output -
(1 row affected)
Server Message:  Number  11518, Severity  20
Sybmail encountered CMC Error:    One or more of the specified recipients were
not found..
(1 row affected)
(1 row affected)
(1 row affected)
All messages sent!
(return status = 1)
```

```
(return status = 0)
(return status = 0)
(return status = 0)
```

This example selects from the *sysusers* table and, for each user name, sends an E-mail message using the **xp_sendmail** command. The script assumes that the names of the users of this database are the same as the logins of the database and of the users on the NT system. If all databases are affected then the DBA can use the *syslogins* table instead of *sysusers*.

> ***Note***: The Error number 11518 arises because one of the users was not configured with an E-mail account.

The mail is sent to each user and can be viewed using the any standard mailer. It is important to size the E-mail infrastructure if your Sybase application might be generating a large amount of traffic on a regular basis. The following example shows a standard mailer:



From	Subject	Received
bueche	Sales are up!	Wed 4/1/98 11...
	Things are looking good. Sales are up! <end>	
bueche	DBMS Application will be down Tomorrow	Wed 4/1/98 11...
	The XYZ application will be down tomorrow for a planned software upgrade. <end>	

2 Items, 2 Unread

Ready | BIZET | bueche | master | Text

For Help, press F1

Figure 6 Using Sybase ASE in enterprise mail environments

TEST APPROACH

Capacity and Performance Considerations

The following example covers a large 2-tier client/server configuration. The ASE configuration file was changed to support 10,000 users. The tests run supported up to 5,000 users. This configuration is a good example of the key issues and parameters settings that should be considered when supporting large user communities.

ASE Parameter Settings

- Total memory was set to 400,000 pages (essentially 800,000 KB),
- The number of user connections was set to 10,100
- The procedure cache was set to 40 percent
- The Number of locks were configured to 20,000
- The DBMS task (or fiber) stack was changed to 64 KB.
- The starting virtual address was changed to 805306368 (which is 30,000,000 HEX)
- The number of engines was set to three (for large concurrent users, the number of engines should be one less than the number of available processors, this is likely to change in future releases)

ProLiant Server Configurations

The system had the following characteristics:

- Proliant 6000 with 4 200-MHz Pentium Pro processors and 3 GB memory
- Single 100-Mbps network card
- Windows NT 4.0 Enterprise Server

TEST RESULTS

The application benchmark tested was based on the standard AS^{3}AP multi-user variant of Client Server Solution's Benchmark Factory. This software allows the tester to vary the number of clients and change things like query content and client think time. The clients were varied from 1,000 to 5,000. The 5,000 clients were spread over 10 Compaq Deskpro 2000 6/180's each with 81 to 128 MB of memory. Each client had a think time based on a uniform distribution ranging from 30 to 45 seconds.

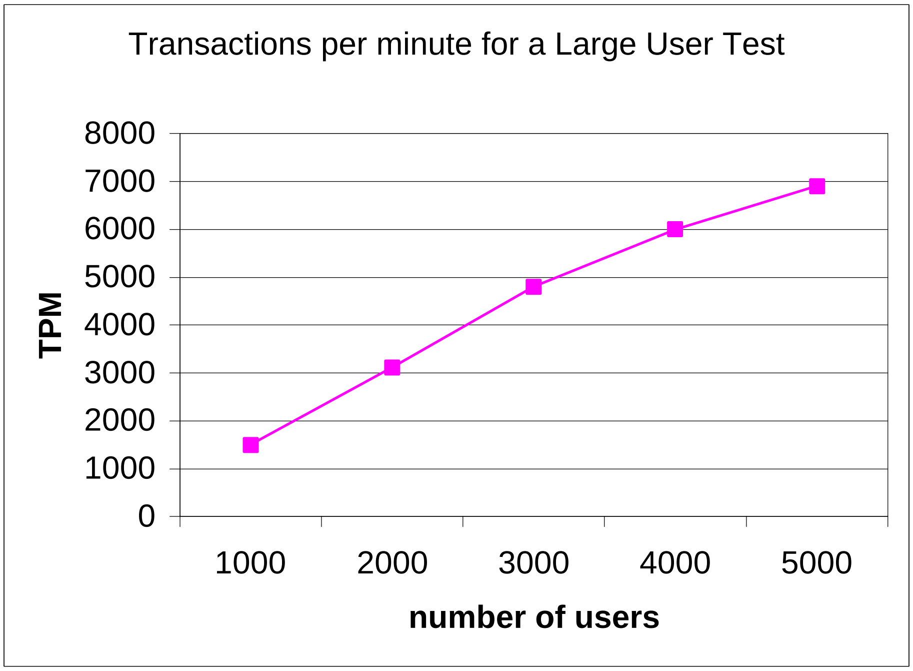


Figure 7 – ASE Performance Results

The benchmark being executed was based on stored procedures, hence, each transaction caused only a single message request/response exchange. This is optimal for large user network traffic. Actual performance of applications with large numbers of users could vary depending on several factors including:

- The number of messages exchanged per transaction and per user: Several ERP applications use cursors on the client side to interact with the DBMS server. This could imply at worst 6 messages exchanged for a cursor open, fetch and close rather than a single pair of messages as for stored procedures. This could have a significant effect on the number of users supported.

  Worse yet, a transaction could consist of many cursor (or separate stored procedure) operations thus multiplying the message load. Stored procedures, on the other hand, can contain many operations and can reduce a transaction down to a single message exchange.

- The size of the messages: If messages are larger then fewer messages can be sent (but the byte throughput is typically much larger). This could arise in Intranet applications that display bitmaps stored as image data in the database. Adaptive Server Enterprise can alleviate this problem by supporting multiple TCP/IP interfaces thus increasing the network capacity of the system.

- The bandwidth that all users have access to: Response time could vary greatly among users with different bandwidth capacity and line quality. Clearly, a user with a 9.6K bps modem access is likely to experience much longer delays than one on a 100Mbps Ethernet backbone.

- The think times of the users: With less think time the large numbers of users will be generating more network traffic. Simple benchmarks typically have little or no think time between transactions, most real applications have large amounts of think time. A good example of think time in the real world is a gas station credit card machine. Since it takes 4 minutes to fill a tank at best one terminal will issue one transaction every 4 minutes. The think time in this case is the gas tank fill time. An absence of a think time in a benchmark could easily limit the number of user connections supported with reasonable response times.

- Network protocols employed: Certain protocols like Named pipes perform poorly over routers because they have to be encapsulated within TCP/IP packets.

- The number of connections established per "user": As mentioned previously, some applications actually establish multiple DBMS connections for each user. So for example, a single user could have 4 separate connections to the database. Typically, only one connection is actively exchanging messages, however, 500 "users" could imply up to 2,000 simultaneous connections.

  In addition, the frequency that connections are established and terminated can also change the performance characteristics of the system. Some applications that spawn multiple connections will have most of their "additional" connections only for a short duration. One way to get around this is an application specific connection cache: don't terminate the session, rather, reuse it for an upcoming connection.

- The workload itself: An application with many Decision Support parallel queries is less likely to have spare CPU and disk capacity for many user connections. In general, the lighter the processing steps of the transactions or queries the more users and user connections one should be able to support.

Its good to review the network and application message profile when setting up any network intensive application. If all things are optimal and there are sufficient system resources then the limits of the number of simultaneous connections could vary anywhere from 2,500 to 5,000 on a 4 way Pentium Pro machine. But this is extremely application specific and should be evaluated for each installation.