



Adaptive Server Enterprise on Egenera BladeFrame platform

A technical white paper
January 2003



Information Anywhere

EXECUTIVE OVERVIEW

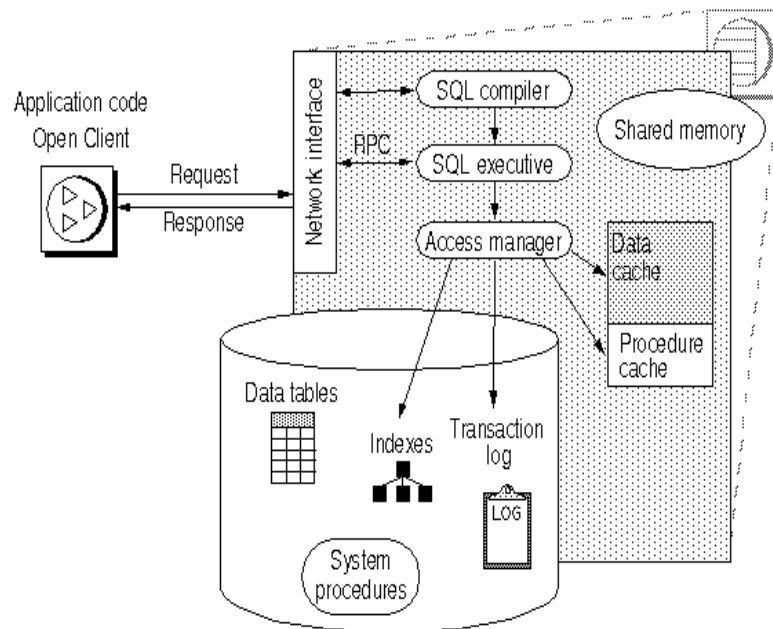
Blade Servers using Intel Processors are modular, efficient and cost-effective systems. They provide virtualized storage and network using a high speed interconnect switched fabric. The management software eases the pain in administering the Blades. Meanwhile, the Linux Operating System has matured in terms of reliability, availability, scalability and manageability. The price/performance of Intel/Linux platform makes a compelling argument to run mission critical applications like Database Servers, Enterprise Resource Planning (ERP) applications.

Sybase ASE is one of the early adopters of Linux OS and released Sybase ASE 11.0.3.3 in 1998. Sybase ASE designed to support the demanding requirements of transaction-intensive, mission-critical OLTP and decision support applications. Sybase ASE scales well from 2-way to 64-way processing environments. ASE's efficient query optimization delivers unsurpassed levels of performance and scalability. The most recent release of ASE, 12.5, builds on this solid foundation with innovations such as handling new data types, dynamic performance management to address unpredictable requirements and enhanced security to protect business data.

Sybase ASE on blade servers reduces the Total Cost of Ownership (TCO) and gives the best Return On Investment (ROI). Sybase ASE is the best scalable, high performing and highly available and easy to manage database engine in the Enterprise class on the Linux/Unix platform.

SYBASE ASE ARCHITECTURE

Sybase ASE runs as an application on top of an operating system and depends solely on the services exported by the operating system to function. It uses operating system services for process creation and manipulation; device and file processing; shared memory for interprocess communication. ASE uses its own thread package for creating, managing, and scheduling tasks inside Sybase ASE engines for running SQL Queries and for System tasks like checkpoint, network handler, house keeper etc.



DYNAMIC PERFORMANCE

With its unique tuning capabilities, ASE can adjust its cache of index pages and data, reduce I/O costs, provide user priorities, control resources at the CPU level, or limit resources for a query or batch. Quick and easy adjustments allow DBAs to reconfigure the system without rebooting.

SECURITY

Row-level storage security in ASE through access rules on columns can control the rows that can be accessed by each user. SSL protocol can be used for integrity checking and encryption mechanisms to protect data from unauthorized access. In conjunction with SSL, digital certificates can be used to identify

and authenticate servers, while clients can be authenticated using the existing username/password mechanism.

MISSION-CRITICAL COMPUTING

ASE continues to support mission-critical systems with many new features including Expanded Row and Data Sizes, Dynamic Configuration, Compressed Backups, Enhancements to the Component Integration Services (CIS), SQL Debugger, Union in Views and Superior Management of Temporary Data.

ADVANCED DATA MANAGEMENT

Sybase ASE is designed to optimize and extend the capabilities of the database application developer by providing new tools for handling emerging e-Business data types. For Advanced XML Information Management, ASE provides a simple, flexible, and powerful XQL query engine to allow dynamic content generation from XML document storage without requiring custom code to handle XML documents. EJB components can optionally be deployed and run inside the database engine, resulting in better performance because of reduced network communication overhead.

MANAGEMENT OF EXTERNAL NON-RELATIONAL CONTENT

Sybase ASE offers an option that provides an easy and cost-effective solution to manage external content. Through proxy tables, all information in an Operating System directory can be accessed. Using already familiar SQL commands, information in Operating System files can be created, deleted, read or revised.

EGENERA BLADEFRAME OVERVIEW

The Egenera BladeFrame was purpose-built to solve the multiple pain points of enterprise-class datacenters by delivering the next generation of computing infrastructure at a lower TCO than current architectures. By integrating processing, networking and management functionality currently dispersed across server hardware, data networks and operating systems, Egenera creates a new architecture, the PAN, which consolidates and simplifies the allocation and management of computing power. The Egenera PAN architecture is a group of autonomous, stateless Processing Blade™ resources redundantly connected via a high-speed, low-latency switched fabric. With features such as two- and four-way symmetric multiprocessing across up to 24 nodes, built-in high availability and dynamic resource repurposing, the system was

designed for mission-critical applications such as databases, application servers and transaction processing.

Virtual servers are created using the Egenera's BladeFrame management software, the PAN Manager, which are called pServers. These are built through software by combining Egenera processor blades (pBlades) with disk, SAN or NAS, and networking resources.

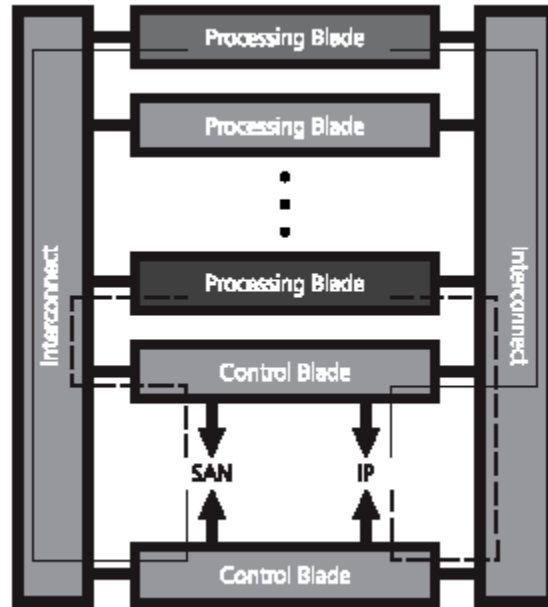


Figure 1: The architecture of the BladeFrame is composed of up to 24 processing blades

AVAILABILITY CLUSTERS

Egenera has High Availability built-in the system. The BladeFrame and its PAN Manager software handle server hardware and software failures.

Sybase ASE takes advantage of this Availability Clustering functionality built-into the BladeFrame platform. Sybase ASE does support both active/active and active/passive failover topologies. During a Sybase ASE instance failure, due to either a hardware or software cause, Egenera facilitates in restarting the failed Sybase ASE over to another pServer. Sybase ASE quickly recovers by replaying in-flight transactions (redoing committed transactions

and compensating uncommitted transactions) thus allowing continuous database processing as long as one pBlade is available.

ACTIVE-PASSIVE AVAILABILITY CLUSTER

ACTIVE-PASSIVE (A-P) option in Egenera bladeframe is done by failing over an application or set of application(s) to another Processing Blade (pBlade) that is inactive. This section describes how-to configure Sybase ASE and RS and use ACTIVE-PASSIVE availability option.

Make sure you install Sybase ASE 12.5 through rpm utility. This will generate a “sybase” script in /etc/init.d with “start”, “stop” and “restart” options.

As superuser create a link in /etc/rc3.d as follows

```
ln -s ../init.d/sybase S99sybase
```

The above script will start all sybase servers (Sybase ASE, Sybase Backupserver, Sybase MonitorServer and Historical Server) when the OS starts and gets into Level 3.

With the above script and link in place, when the primary pBlade goes down the PAN Manager will failover the application to the pBlade configured as a failover pBlade. The pServer boots at that time (PASSIVE) and restart Sybase servers, which in turn will do fast recovery and ready to accept new connections. During the time of failover and recovery the client connections must retry. The client interfaces file entry should have the connect retry count (number of times to retry connection) and connect retry delay (the time delay between each retry) set properly.

ACTIVE-ACTIVE AVAILABILITY CLUSTER

ACTIVE-ACTIVE (A-A) option in Egenera bladeframe is done by failing over an application or set of application(s) to another pServer that is active. This section describes how-to configure Sybase ASE and RS and use ACTIVE-ACTIVE availability option. The following is just a sample on how it can be done. Please refer to the Egenera manuals for details.

Using Egenera failover toolkit -

- Create failover policy
fop -c -L "ASE fail over policy" -F yes -a <failover_egenera_node> -O Sybase(LPAN)/fopsybase
- Configure MON Server
hmonsvr -c Sybase/<primary_egenera_node>
- Create Health Monitor
Hmon -M "dbmon -dbclasspath
\$SYBASE/jConnect-5_5/classes/jconn2.jar \
-d com.sybase.jdbc2.jdbc.SybDriver \
-l jdbc:sybase:Tds:<ip_addr>:<port#> -Usa -P"" -t 5" \
Sybase/mysybmon
- Define IP Address:
ip -c -I 192.168.2.165(IP address of db2) -A 1
Sybase/sybase_ase
- Define a file system resources
fs -c -M /opt/sybase-12.5 -D (5.0.10.0) -P 1 -A rw Sybase
- Configure an application for PAN Manager control
app -c -L Sybase_ASE -P db2 -N /etc/rc.d/init.d/sybase -a / -A
yes -I sybase_ase -H mysybmon -F fopsybase
Sybase/sybase(lpan/application name)

Things to consider –

- All Sybase ASE instances setup to fail over in an ACTIVE-ACTIVE should be using unique disk devices and mount points (if using file system files as Sybase ASE devices) across the multiple nodes that is configured to be failing over. If using raw devices make sure the raw device bindings are also unique.
- Make sure the **shmmax** and other Operating System values are uniform across the nodes.
- In an ACTIVE-ACTIVE failover scenario, the root partition “/” is not failed over. Make sure that SYBASE release area is available in the failing over node (if it is not already available). The interfaces file also should be available to start the Sybase servers in the failed over node.

DATABASE APPLICATIONS USING BLADE SERVERS

Blade Servers lowers TCO and raises the ROI with the scalability, availability, and manageability.

Unlike scaling web servers to N nodes, scaling a Database Engine to more than one node or machine is not a trivial task. Database Engine does manage updates/insert/deletes/select against a database store. They provide atomicity, concurrency, isolation consistent, and durability by doing locking, logging and recovery. Even though clustering/distributed computing provides high availability it provides little scalability. Providing seamless/transparent functionality and linear scalability is a challenge in terms of manageability issues (data partitioning, backup, recovery) or scalability issues (inter-node synchronization).

In order to get linear or near linear scalability in distributed systems, it is necessary to partition the application and data. This holds true for all currently available systems, unless the data is only updated in one place. For Sybase ASE there are quite a few options to achieve this by leveraging Sybase ASE Federated/Distributed Services (Component Integration Services), ASE Replicator and Sybase Replication Server.

1. WORKLOAD PARTITIONING

This is by far the most common approach to scale beyond one machine, to separate transaction processing oriented workload from decision support or reporting. DSS applications typically only read data and don't update. Since the data is updated in only one place, no action needs to be taken to ensure consistency, and to copy the data or subsequent updates there are a number of alternatives:

- Sybase Replication Server
- Backup copies
- SAN-features

Replication, whether ASE Replicator or Sybase Replication Server, keeps the DSS node(s) up to sync with the ongoing data updates in near real-time. Sybase Replication Server also provides excellent transformation capabilities to tailor the data model for the workload.

Backing up the OLTP system to load the backups on other nodes has been practiced for many years. Since ASE backups don't interfere with system performance, they are very fast and scalable, and always provide a time consistent copy, it is a simple and convenient way of getting a DSS or reporting nodes in place and

updated. With the option `standby_access` on the `dump tran` and `online` commands, subsequent transaction log backups can be loaded and used parallel to the running OLTP system. Also, it is not uncommon that the required point-in-time is well known to users and applications, especially with non-24-7 systems, further paving the path for choosing daily refreshes by backups as the easiest approach.

Disk subsystems often provide the ability to copy data internally, i.e. without the applications being aware or affected by the operation. This by either a straightforward copy or by splitting a mirrored pair. By starting another ASE with the “-q” option, `dataserver -q`, on the split copy it can be used instead of database backups, for e.g. reporting, and if required refreshed by database transaction log backups.

In this architecture routing of requests are done either by the application connecting to the DSS ASE for querying or by maintaining proxy tables in the OLTP ASE pointing to the physical tables in the other ASE(s).

2. FUNCTIONAL/DATA PARTITIONING

In functional partitioning scheme, different functions within an organization, e.g. users with the financials, production and sales departments may be mostly using their own data, and the common data is accessed for read but updated infrequently. In data partitioning scheme within a functional group/organization, different regions work with data unique to that region and require only read-only access to global data. In this case there are several options for partitioning:

- Origin, e.g. source IP address
- User id during web or application server login phase
- Application level lookups in meta-data to use a particular database server
- Use the functionally partitioned database servers

If data is updated in only one place and then needed in read-only copies at the other nodes replication is a very good option, ASE Replicator or Replication Server. Sybase ASE also has Component Integration Services to get access to remote data and union in views to present this as a single collection and global view to users and applications.

Given a particular IP address, subnet mask, etc. of course a router can direct the request to the right web or application server. This can also be done on the Sybase protocol level, TDS, by using OpenSwitch, which only routes Sybase client and provides additional transparency regarding the underlying database server layer.

Given a user id presented in the login phase or meta-data can be used to direct the user to the right ASE, and kept there not only for the duration of the session but until architecture changes prompts re-routing.

In data partitioned topology, the application provides a set of rules applied to the data to determine which database it should reside on. Sybase ASE provides functionality like Component Integration Services, Union in View options to provide a seamless transparent single cohesive data set on top of the distributed data sets.