

::ISUG

Sybase EAServer

techcast
series

Tips & Tricks

This live, interactive seminar will begin shortly

For audio, please dial:

Toll Free within the U.S. & Canada: 1-888-459-9364

International: 1-210-234-6662

Passcode: ISUG

::ISUG

Sybase EA Server

techcast
series

Tips & Tricks



Bryan Enochs
Moderator

NA Regional User Group Director
International Sybase User Group

<http://www.isug.com>

::ISUG

Sybase EAServer

techcast
series

Tips & Tricks



Javier Cuerva

Presenter

Senior Technical Evangelist

cuerva@sybase.com

- **Tuning Tricks**
- **Design Tricks**
- **Troubleshooting Tricks**
- **Q & A**

- **Some settings must be made on the OS to improve performance.**
Avoid "connection refused" errors by clients
nnd command deals with tcp/ip internet protocols
 - `nnd -get /dev/tcp tcp_conn_req_max_q`
 - `nnd -get /dev/tcp tcp_conn_req_max_q0`Should be set to at least 1024
 - `nnd -set /dev/tcp tcp_conn_req_max_q 1024`
 - `nnd -set /dev/tcp tcp_conn_req_max_q0 1024`No reboot is required after setting the command
- **tcp_conn_req_max_q**
The default maximum number of pending TCP connections for a TCP listener waiting to be accepted by [accept\(3SOCKET\)](#)
Default is 128
- **tcp_conn_req_max_q0**
The default maximum number of incomplete (three-way handshake not yet finished) pending TCP connections for a TCP listener.

- **Increase the number of allowable open files.**

Can cause refused connections or increased wait time.

Edit the /etc/system file

Number of fd a process can open, soft limit, default is 256

When to change : when the default number of open files for a process is not enough

Soft Limit on file descriptor a single process can open

This means the program may not use the setrlimit () primitive

- set rlim_fd_cur=4096

Hard Limit on file descriptor a single process might have open

- set rlim_fd_max=4096

Reboot is required

Flow control means limiting the number of threads that can do real works at the same time.



V.S.



Handler threads are competing with each other for resources and creating traffic jams and eventually the response time of each thread degrades to an unaccepted degree.

com.sybase.jaguar.server.flowcontrol.http

Indicate if HTTP requests are subjected to flow control

com.sybase.jaguar.server.flowcontrol.maxexethreads

A positive integer, to indicate the maximum number of concurrent execution threads allowed at one time.

com.sybase.jaguar.server.flowcontrol.iiop

Indicate if IIOP requests are subjected to flow control

com.sybase.jaguar.server.flowcontrol.trace

Trace flow control activity

HTTP HANDLER ALGO

For a new http request do :

```
Server_flowcontrol_proceed(FLOWCONTROL_HTTP);
ret = http_process_request(request);
Server_flowcontrol_done(FLOWCONTROL_HTTP);
```

Proceed :

If curexethreads >= maxexethreads
make current thread wait

IIOP HANDLER ALGO

For a new http request do :

```
Server_flowcontrol_proceed(FLOWCONTROL_IIOP);
//process here the iiop request
Server_flowcontrol_done(FLOWCONTROL_IIOP);
```

Done:

If exists waiting thread
wakeup current thread

- **Log4J and JDK 1.4 java.util.logging integration**
- **The built-in EAS subsystem:**
 - The ability to configure log levels
 - Messages below a specified level of severity are discarded.
 - Support for different logging configurations in the debug and production servers.
 - Optional archiving and compression of previous log file versions.
 - More control over message formatting.
- **You can still use**
 - System.out.println for JAVA
 - ErrorLogging for PB
 - JagLog for C++
 - IJagServer.writeLog for ActiveX
- **You can use the EAS Logging System in your components**
 - Create a log profile or derive one from the log profile available (debug, prod, debug_jdk, prod_jdk, debug_l4j, prod_l4j) see Repository\LogProfile

- **Improves Performance for EJB -> EJB Communications.**
- **Server level Property**
`com.sybase.jaguar.server.lwc = true | false`
- **Component level Property.**
`com.sybase.jaguar.component.lwc = true | false`
- **EAServer activates the lightweight container for EJB-to-EJB calls as long as all of the following are true:**
 - The LWC is enabled for the server and the called (target) component.
 - The calling (source) component uses an EJB reference or EJB local reference to make the call.
 - The LWC is compatible with the transaction and security properties of the calling (source) and called (target) components.

- **Significant performance improvement with "Required" or "Requires New".**
- **Object references cannot be marshalled**
 - A calls B and B returns a reference on C (solution disable LWC for C)
 - Disable LWC for components returning or using object references as parameters.
- **Debug info**
 - `com.sybase.jaguar.server.lwc.debug = true | false`

- **Use local interface instead of remote interface**
 - Causes less marshalling (no network IO)
 - Parms are passed on the stack.

- **Web Application Caching**

 - Static Page Caching

 - Simple Caching

 - Dynamic Page Caching

 - Partial Caching (New in 5.0)

- **EJB.**

 - Improve performance by avoiding the loading of instance state at the start of each transaction.

- **Turn on Static caching**

- com.sybase.jaguar.server.http.cache.debug

- com.sybase.jaguar.server.http.cache.enable

- com.sybase.jaguar.server.http.cache.exclude-files=

- com.sybase.jaguar.server.http.cache.size

- com.sybase.jaguar.server.http.cache.timeout

- com.sybase.jaguar.server.http.cache.webapps.exclude-files

- **Saves static HTML information**

- Html files, Gif files, Jpeg files

- Etc.

- **Implement Java side Servlet cache which will cache outputs of Servlet/JSP in java core memory.**
- **Fast response for Servlets or JSPs that the output will not be changed during the timeout period.**
`com.sybase.jaguar.servlet.javacache.enabled`
- **Default is false**
`com.sybase.jaguar.servlet.javacache.session`
Specified how session, cookies should be treated.
- **Keep – Attempt to preserve valid sessions**
- **Create – Preserve valid session and create a new session if session is invalid**
- **No – No session support. Do not check or return "Set-Cookie" session header.**

`com.sybase.jaguar.servlet.javacache.maxsize`

- **The value is the number of K bytes indicates the limit of the content size.**
- **If a response is > than the cache size, this is not cached**
 - Default is 8k
 - Limit – 100k

`com.sybase.jaguar.servlet.javacache.timeout`

- **The timeout in seconds.**
 - Default 60
 - ≤ 0 means never expired.

- **Partial page caching allows you to cache parts of a response.**
- **Supported by a tag library for use in JSPs, and a public API for use in servlets.**
- **Uses** CacheTags.jar

```
<%@ taglib uri="http://www.sybase.com/EAServer/cachetags.tld" prefix="ct"%>
```

- **To cache a portion of a page, surround it with this tag, as in:**

<ct:cache attributes>

... page content ...

</ct:cache>

- **Where caching attributes can be**

Parameters, session attributes, headers, timeout, scope, name of cachelet ...

- **The API is implemented by class CacheManager**

Create

Get Data From Cache

Places Data in Cache

Flush

- For update transactions, may be able to avoid any SQL “select” statements, while still having to execute a SQL “update” statement.
- For read-only transactions, may be able to entirely avoid the execution of SQL statements.
- For CMP entity finder methods that return collections, may be able to run one query instead of N+1 queries for finders returning a collection containing N entities.

- Frequency of re-use of objects with same key, e.g. how often does a new transaction use the same entity as a previous transaction for which an entry is still present in the object cache.
- Ratio of update to read-only transactions.
- Size of data set (e.g. database table).
- Size of object cache.
- Configured timeout for cache entries.

- Interface `CtsComponents::ObjectCache`
(basic operations: `init`, `find`, `get`, `put`, `remove`)
- Component `CtsComponents/ObjectCache` (built-in implementation) in `dispatcher`, `ObjectCacheImpl.cc`
- Configurable – each entity/stateful component can have its own cache configuration. Caching is disabled by default.
- Pluggable - partners and end users can write their own cache implementations (components).

- Built-in component is implemented in C++ for
- Two reasons:
 - Maximum performance
 - Cache data is not visible to Java VM garbage collector. This allows extremely large caches, e.g. 500Mb.
- Cache implementations must use C++ or Java (requires the use of the “Bind Object” option).

- **Drivers to improve overall performance of CMP EJBs**

- Allowing updates to be deferred to the end of each transaction and sent together as a command batch..

- Reducing network round trips between the database server and EAServer.

- Supports automatic creation of semi-temporary stored procedures.

- **Sybase Driver** : `com.sybase.ejb.cmp.SybaseDriver`

- **Oracle Driver**: `com.sybase.ejb.cmp.OracleDriver`

- **Lots of new Params... Check the documentation**

- **Deployment of PB Components creates**

%JAGUAR%

Repository

Component

<Package Name>

<Component Name>.props

<Component Name>

C<n>

<one or more PBDs implementing component>

- **Sharing Class Group Loaders**

com.sybase.jaguar.component.pb.librarylist

Has a value of \$PBD1.PBD;\$PBD2.PBD;\$PBD3.PBD;...

A unique instance of a class group loader is assigned to each distinct occurrence of a library list

The default will cause a unique library list for each PB component deployed because of the substitution that occurs

- **Sharing class group loaders - solution.**

Include all PBLs containing the NVOs that are sharing resources in a single, combined library list.

Deploy each component as you normally just ensure you use the same deployment options in terms of PBL consolidation.

Use jaguar manager to modify the `com.sybase.jaguar.component.pb.librarylist` property in the properties file of each component to be the exact same list by referencing a fully qualified path to the constituent PBDs.

Refresh the components.

Backup the changes to the components' properties files.

- **Increase Listener settings**

- **`com.sybase.jaguar.listener.http.conn.maxrequests`**

Specifies the maximum number of HTTP requests to service before closing each connection

It must be greater than the number of iterations of each client will run avoid unnecessary close of socket– key when doing stress testing.

If set to low can cause invalid server response

- **JSP Settings**

Do not check if the JSP is outdated or not

- **`com.sybase.jaguar.webapplication.jspc-interval=-1`**

- **Turn off HTTP logging**
- **Insure the number of threads is set correctly**
`com.sybase.jaguar.server.http.maxthreads`
`com.sybase.jaguar.server.maxconnections`
`com.sybase.jaguar.server.maxthreads`

- **Tuning Tricks**
- **Design Tricks**
- **Troubleshooting Tricks**
- **Q & A**

- **Pooling objects**

Programmatic pooling is available for objects implementing the COM ObjectControl or EJB 0.4 ServerBean interface.

- **canBePooled for PB/COM**
- **canReuse for CORBA objects**
- **Called if the pooled flag is not set.**

Declarative pooling is available via “Pooling” check-box in Component Properties.

A min and max pooling value can be set to help manage memory.

- **Develop small components**

one ancestor object gets inherited and code specific logic

Most components can be developed with two pbls

- **Keep library list small.**

Publish one component per application.

This keeps the size of the component down

PB does not read the class definition in one shot, and access the PBLs several times aka on demand, to get information on helper classes or services.

- Create all datastores in the components constructor event and initialize the dataobject.

- Don't over design too many inter-component calls kill performance and take up too many resources.
- Don't use PB components as services if the component will run in a loop and you need to access the service code from another component. Use ThreadManager to develop more robust services.
- Use JagSleep primitive if need to run a loop never the sleep method

- **Untuned SQL**
Slow Performance of the SQL within the DB
- **Mis-use of the connection caches**
Not returning connections to the cache
 - **Causes more JCM_FORCE connections**
Use of JCM Wait
 - **Makes client think the server is hung**
- **Lots of I/O to the log**
Provide a switch that will turn debugging on/off
Write to the log via the MessageService
 - **Time delay in when it is written to the log.**

- **Tuning Tricks**
- **Design Tricks**
- **Troubleshooting Tricks**
- **Q & A**

- **PB Version**

Windows:

- **GetFileVersionInfo** window API. See sybase.com for the white paper

Solaris:

- **strings \$JAGUAR/lib/libpbvm70x.so | grep Version**

New EAS Manager in 5.0 gives the same information now.

- **Turn on the verbose mode**

Shows all of the classes that are loaded and when.

- **com.sybase.jaguar.server.jvm.verbose=true**

- **EAS 5.0 New Feature**

Allows to export to a text file

- **Server Settings (listeners, Wide Server configuration, Packages, Connection Caches, Roles ...)**
- **Server logs (console logs, file logs, http request log ...)**

Generation from EAS Manager

- **Server/Your_Server/Export Diagnostic Log...**

- **Getting the IIOp stack**

Tells all of the IIOp information that is sent and used by the server

- **com.sybase.jaguar.server.iio.log**

Client application issues, such as login failures

Component issues:

- **Method invocation errors**
- **Result sets not returned as expected**
- **Trouble connecting to the target database**
- **Intercomponent call errors**

Warning: This will create a LOT of output. Don't leave on except for a short time.

- **JagRepair**

Read only version of the server

Allows you to make configuration changes that may be causing the server to crash (OTS, Connection Cache, Service Components, etc.)

How do I start it

- **Serverstart –servername JagRepair**

How do I connect to it?

- **UserName: jagadmin**
- **Hostname: localhost**
- **Port Number: 9000**

- **Used by EAServer to find and load object**

java.classes property

Found in:

- **Servers (4.0)**
- **Packages**
- **Components**
- **Connectors**
- **Servlet**
- **Web Application**

Defines the classes that are required to load a particular object

Defines the jar file(s) that are needed (in addition to the bootclasspath) for finding classes

- **When comparing classes they need to be loaded at the same level**

Example

- **ServletA creates object Foo**
- **EJB_A creates object Foo**
- **ServletA passes it's Foo to EJB_A as a parm. EJB_A tries to use Foo from ServletA**

Resolution: Have the server class loader load Foo

- **Most functionality in the server has a debug flag**

Documented and undocumented

- **com.sybase.jaguar.<feature>.debug=**

Examples

- **com.sybase.jaguar.server.http.cache.debug**
- **com.sybase.jaguar.component.debug**
- **com.sybase.jaguar.servlet.debug**

Will impact performance

- **Tuning Tricks**
- **Design Tricks**
- **Troubleshooting Tricks**
- **Q & A**