

100,000 Concurrent User Benchmark Study FAQ

1. What possible improvements to the product were found from conducting the benchmark?

We don't comment on the details as to future product enhancements, but we have a number of findings as to how we can improve our Content Server with respect to memory and CPU utilization.

2. What technical challenges did the team encounter in attempting the benchmark?

Well, we have found with most benchmarks not even the best planned are not without problems. During the middle of the benchmark a whole rack of servers lost power in the middle of a test during a huge snow storm in Seattle. Unfortunately they were in a non recoverable state upon attempting to power up. We ended up having to rebuild and reconfigure.

One interesting note, after the loss of the blade server chassis due to power interruption the EMC Clariion was still operational and all data was intact.

Also, with the high rate of traffic that we were generating on the HP Superdome (database tier) we found working with an HP engineer that we were overwhelming the single NIC card 60,000 packets/sec and had to add a total of 5 NICs on the Superdome to support the traffic from each Content Server. We also used a Intpolicy and set affinity for all NICs to specific CPUs which were local to the IO chassis in which the NIC resided.

A general rule of thumb is to add an additional NIC when you reach 30,000 packets/sec per NIC card.

With the large number of machines required to produce the load it was challenging managing the sheer number of physical pieces of hardware. A remote desktop product from Microsoft made it easier viewing and managing the entire configuration from a single desktop.

We relied on expertise from Microsoft, HP, and EMC to configure and run this benchmark successfully. For every tier we had different experts that managed the hardware, software, network, and storage. We relied on each other to provide the given domain expertise when various issues arose.

3. Where were the bottlenecks in performance and how did you address them?

Here are some of limits that we ran into during the benchmark.

Sessions: We needed to limit the number of session per content server to 400 to control the memory pressure on the content server.

Even with Windows 64 bit OS we still limited to how much memory was addressable by the content servers as they were running in 32 bit emulation mode.

NIC Cards: The NIC card on the database were overwhelmed when the packets/sec rate reached 60,000 packets/sec, so we had to add more cards and set affinity to specific CPU to help balance the load.

Loadrunner: Vusers per Loadrunner Controller is limited to 65,534 users after that we need to configure a second controller.

Database: Having a machine that had 64 processors and 256GB memory definitely reduced the chances of any bottleneck at the database tier. However, we did need to ensure that all transactions and their associated queries were tuned appropriately.

4. Did your team use a special build of the content server for the benchmark to support SQL Server 2008?

We used 6.5 SP1 Content Server plus the following pre GA modules :

- cfsConfigurationProgram.exe
- Server_Configuration_Program.exe
- dmadmin.dll
- dmdbtest.exe
- dmserver_v4.dll

SQL Server 2008 is supported with 6.5 SP2 of Documentum Content Server

5. Could you clarify what EMC Documentum needed to do to support HP Polyserve File Clustering Software?

Nothing. Polyserve, VERITAS, Ibrix and others all allow customers to set up file clustering support for a set of servers with specific OS to support the sharing of common data. We have no code in our Documentum Content Server written specifically for this technology. The OS and the file clustering software look after the sharing. We just had to install and configure the file clustering software for the configured network and SAN topology. The support for file clustering comes from the software vendor that sells this technology.

6. Can you please clarify which tier ran 64bit Windows Server?

Application Server Tier

- Microsoft Windows Server 2008 32 bit Enterprise SP1, Build 6001
- Apache Tomcat 5.5.25
- Java 1.5.0_14

Content Server Tier

- Microsoft Windows Server 2003 64 bit Enterprise SP2, Build 3790
- EMC Documentum 6.5 SP1 Content Server 32 bit with updates to support Windows SQL Server 2008. Running in 32 bit emulation mode to allow us to address more memory and run two instances of the Content Server per physical machine.
- HP PolyServe 3.6.0 Clustering software (PolyServe at the time of the test did not support Windows 2008 hence the reason we ran on Windows 2003.)

Database Tier

Microsoft Windows Server 2008 64 bit for Itanium Based System Version 6.0.6001 Service Pack 1 Build 6001, Microsoft SQL Server 2008 64 bit Enterprise Edition (Previous studies that we had conduct in the past with Documentum 5.3 with Microsoft had shown that we could achieve better performance using a 64 bit OS and Database versus a 32 bit offering. (See Case Study <http://www.microsoft.com/casestudies/casestudy.aspx?casestudyid=4000000559>)

7. Noticed that the benchmark system used SQL Server 2008. Is Content Server 6.5 SP1 support this version of SQL Server?

SQL Server 2008 is supported with 6.5 SP2 of Documentum Content Server

8. Was this done using the SQL Server 2005 compatibility mode?

No. The value of the setting on the database was as follows:

COMPATIBILITY_LEVEL = 100 (100=SQL Server 2008)

9. What was the reason for choosing SQL 2008 since it is not officially supported by D6.5 SP1?

Part of doing a high-water benchmark with partners is to prove out new technology. Microsoft had worked with us in the past to conduct benchmarks on Documentum 5.3 with SQL Server 2000 and 2005. Early in 2008, they approached us asking if we were interested in benchmarking Documentum on SQL Server 2008. At this time 6.5 SP1 was already out, so we worked with our Server team to create the special modules required to add on top of 6.5 SP1 to support SQL Server 2008 for this endeavor. This meant that we had to run a number of tests in our internal labs to make sure our product ran well on this pre GA code line before taking our code to Microsoft Enterprise Engineering Center in Redmond to run this benchmark.

Given these efforts Documentum Content Server D6.5 SP2 will support Microsoft SQL Server 2008.

10. Storage

The clustering is not a supported configuration with us. Is this again a thing chosen by our partners HP/Microsoft or was there a technical reason for doing it this way?

It seems to me that it would have been more “real world” if the storage had been NAS connected without the clustering software in the mix. Using multiple

data movers to get the necessary transfer bandwidth. I would appreciate any comments and feedback you could provide on why that configuration was not chosen.

Actually, it depends on who you talk to in the field. We have worked with customers that use similar file clustering software and fiber attached SAN storage similar to our configuration. It all depends on the customer's needs and budget as to what solution is going to be optimal for their production needs.

There is nothing from the file clustering software perspective for EMC to support. This is software that interacts with the operating system to allow a set of servers to share common data. We have Documentum customers in the field today that have been using file clustering software in the field to manage their needs for some time now.

We have tested clustering content servers using NAS in our labs and have found performance less than optimal for our needs. In addition, NAS is not recommended as a storage solution for FAST indexing servers.

11. Java

Although it is nice to see that we were able to do this with Java 5 and an older version of Tomcat, I am curious as to why Java 6 was not chosen for this given all the performance and monitoring/tuning enhancements that were made to Java 6. This also lends itself to asking why you used an older version of Tomcat that doesn't take advantage of the new java platform and in itself is much improved over Tomcat 5 versions.

When benchmarking there is always a tradeoff between the latest product on the market and what has been lab tested many times. In this case we chose to stick with an older version of Tomcat and Java given that we already had a number other newer version of software that we had to learn and manage for this benchmark.

12. Would it be possible to release the details of the configuration so that others can learn how to tune for performance?

Application Server

The following Java command line options were used:

```
set JAVA_OPTS=%JAVA_OPTS% -Xms1024m -Xmx1024m -XX:ParallelGCThreads=2  
-XX:NewSize=256m -XX:MaxNewSize=256m -Xnoclassgc -XX:SurvivorRatio=14 -  
Xloggc:%CATALINA_HOME%\logs\gc.log
```

Dfc.properties on app server

```
dfc.docbroker.host[0]=cs-01  
dfc.globalregistry.repository=EMCSQL64  
dfc.globalregistry.username=dm_bof_registry  
dfc.globalregistry.password=GrRNPhLJrkoTDAZE0RGJow\=\=  
#vb  
dfc.session.max_count = 25  
dfc.session.pool.enable=true
```

Network parameter changes

added TcpTimedWaitDelay to 30 seconds and MaxUserPort to 65534

Other tuning on the Application Server

- turned off firewall
- bumped page file to 4095
- turned off uac
- turned off IE enhanced security
- Updated routes so all app servers are good now. DNS Reverse lookup.

Content Server

Here are the settings used during the benchmark:

Two instances of Content on each single physical machine (32 bit emulation on 64 bit OS)

server_cs-01_EMCSQL64.ini

```
##### DOCUMENTUM CONFIGURATION INFO FILE
##### © 1994-2008 EMC Corporation. All
rights reserved# Version 6.5 of the Documentum Content Server.## A generated
server init file for the Documentum Content Server.# This file was generated on,
Wed Dec 17 18:54:19 PST 2008, by user EMCAdmin.#
```

```
[SERVER_STARTUP]
```

```
docbase_id = 777
```

```
docbase_name = EMCSQL64
```

```
server_config_name = cs-01_EMCSQL64
```

```
database_name = EMCSQL64
```

```
database_conn = EMCSQL64
```

```
database_owner = EMCSQL64
```

```
database_password_file = C:\Documentum\dba\config\EMCSQL64\dbpasswd.txt
```

```
service = Dm_EMCSQL64
```

```
user_auth_target = CHILD
```

```
install_owner = EMCAdmin
```



```
# A boolean flag specifying whether user entries and groups list are synchronized from db.
upd_last_chg_time_from_db = F
# Interval, in second, for checking session user's status.
check_user_interval = 0
# This controls the number of concurrent users that can# be connected to the server at any
given time.# Set sessions low enough to keep Documentum process memory under 2.3GB
which seems to be upperbound of error free processing. Sessions here also translate into
1 to 2 sessions on the DB. Consider fewer sessions will result in more DB activity do to
rebuilding cache more often. Have as many sessions as you can fit in memory and be error
free.
concurrent_sessions = 400
#client_session_timeout = 3
deferred_update_queue_size=2048
# This controls the dmbasic method server.#vb
method_server_enabled = F
method_server_threads = 1
# A boolean flag specifying whether all Index Agent(s) configured for this repository needs to
started automatically during server startup or not#vb
start_index_agents = F
use_estimate_search = T
preserve_existing_types = T
enforce_four_digit_year = T

[DOCBROKER_PROJECTION_TARGET]
host = cs-01
port = 1489
#Comment out proximity set by installer and let it
default#proximity=9001#[DOCBROKER_PROJECTION_TARGET_1]#host =#port
=#proximity =#host=cs-05#port=1489#proximity=9010
```

server_cs-01_EMCSQL64_2.ini

```
##### DOCUMENTUM CONFIGURATION INFO FILE
##### © 1994-2008 EMC Corporation. All
```

rights reserved# Version 6.5 of the Documentum Content Server.## A generated server init file for the Documentum Content Server.# This file was generated on, Wed Dec 17 18:54:19 PST 2008, by user EMCAdmin.#

[SERVER_STARTUP]

docbase_id = 777

docbase_name = EMCSQL64

server_config_name = cs-01_EMCSQL64_2

database_name = EMCSQL64

database_conn = EMCSQL64

database_owner = EMCSQL64

database_password_file = C:\Documentum\dba\config\EMCSQL64\dbpasswd.txt

service = Dm_EMCSQL64_2

user_auth_target = CHILD

install_owner = EMCAdmin

A boolean flag specifying whether user entries and groups list are synchronized from db.

upd_last_chg_time_from_db = F

Interval, in second, for checking session user's status.

check_user_interval = 0

This controls the number of concurrent users that can# be connected to the server at any given time.# Set sessions low enough to keep Documentum process memory under 2.3GB which seems to be upper bound of error free processing. Sessions here also translate into 1 to 2 sessions on the DB. Consider fewer sessions will result in more DB activity do to rebuilding cache more often. Have as many sessions as you can fit in memory and be error free.

concurrent_sessions = 400

#client_session_timeout = 3

deferred_update_queue_size=2048

This controls the dmbasic method server.#vb

method_server_enabled = F

method_server_threads = 1

A boolean flag specifying whether all Index Agent(s) configured for this repository needs to started automatically during server startup or not#vb

start_index_agents = F

use_estimate_search = T

```
preserve_existing_types = T
enforce_four_digit_year = T
```

```
[DOCBROKER_PROJECTION_TARGET]
```

```
host = cs-01
```

```
port = 1589
```

```
#Comment out proximity set by installer and let it
default#proximity=9001#[DOCBROKER_PROJECTION_TARGET_1]#host=cs-
05#port=1489#proximity=9010
```

DFC.properties

```
dfc.data.dir=C\:/Documentum
```

```
dfc.search.ecis.enable=false
```

```
dfc.search.ecis.host=
```

```
dfc.search.ecis.port=
```

```
dfc.tokenstorage.dir=C\:/Documentum/apptoken
```

```
dfc.tokenstorage.enable=false
```

```
dfc.docbroker.host[0]=cs-01
```

```
dfc.docbroker.port[0]=1489
```

```
dfc.docbroker.host[1]=cs-01
```

```
dfc.docbroker.port[1]=1589
```

```
dfc.security.keystore.file=C\:/Documentum/config/dfc.keystore
```

```
dfc.globalregistry.repository=EMCSQL64
```

```
dfc.globalregistry.username=dm_bof_registry
```

```
dfc.globalregistry.password=GrRNPhLJrkoTDAZE0RGJow\=\=
```

Content Server Tuning

--Turn off docbase jobs

update dm_job_s set is_inactive=1 where is_inactive=0;

--Turn off dm_audit_user and auditing.

update dmi_registry_s set is_audittrail=0 where event='dm_default_set';

Updated server.ini to stop method servers, index agents, set proximity.

updated dm_server_config_s as below to stop workflow and set concurrent_sessions.

update dm_server_config_s set concurrent_sessions=450

update dm_server_config_s set wf_agent_worker_threads=0

JBOSS startMethodServer.cmd file change --> This we did because method server was running out of memory.

set NOPAUSE=true

set USER_MEM_ARGS=-Xms1200m -Xmx1200m -XX:ParallelGCThreads=2 -
XX:NewSize=384m -XX:MaxNewSize=384m -Xnoclassgc -XX:SurvivorRatio=14
-Xloggc:gc.log -XX:PermSize=64m -XX:MaxPermSize=256m -Xss256k -XX:
+DisableExplicitGC -Xrs

C:\Documentum\jboss4.2.0\bin\run -c DctmServer_MethodServer -b 0.0.0.0
%JAVA_OPTIONS%

Database Server

Database Server settings

-- SQL used to create benchmark database

USE [master]

GO

/***** Object: Database [EMCSQL64] Script Date: 12/17/2008 17:43:21 *****/

CREATE DATABASE [EMCSQL64] ON PRIMARY

(NAME = N'EMCSQL64', FILENAME = N'E:\SQLDATA\EMCSQL64.mdf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB),

(NAME = N'EMCSQL64_1', FILENAME = N'F:\SQLDATA\EMCSQL64_1.ndf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB),

(NAME = N'EMCSQL64_2', FILENAME = N'G:\SQLDATA\EMCSQL64_2.ndf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB),

(NAME = N'EMCSQL64_3', FILENAME = N'H:\SQLDATA\EMCSQL64_3.ndf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB),

(NAME = N'EMCSQL64_4', FILENAME = N'I:\SQLDATA\EMCSQL64_4.ndf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB),

(NAME = N'EMCSQL64_5', FILENAME = N'J:\SQLDATA\EMCSQL64_5.ndf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB),

(NAME = N'EMCSQL64_6', FILENAME = N'K:\SQLDATA\EMCSQL64_6.ndf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB),

(NAME = N'EMCSQL64_7', FILENAME = N'L:\SQLDATA\EMCSQL64_7.ndf' , SIZE = 13312000KB , MAXSIZE = UNLIMITED, FILEGROWTH = 102400KB)

LOG ON

(NAME = N'EMCSQL64_log', FILENAME = N'M:\SQLLOGS\EMCSQL64_log.ldf' , SIZE = 205824KB , MAXSIZE = 2048GB , FILEGROWTH = 102400KB)

GO

-- 100 = SQL Server 2008

ALTER DATABASE [EMCSQL64] SET COMPATIBILITY_LEVEL = 100

GO

```
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [EMCSQL64].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
```

```
ALTER DATABASE [EMCSQL64] SET ANSI_NULL_DEFAULT OFF
GO
```

```
ALTER DATABASE [EMCSQL64] SET ANSI_NULLS OFF
GO
```

```
ALTER DATABASE [EMCSQL64] SET ANSI_PADDING OFF
GO
```

```
ALTER DATABASE [EMCSQL64] SET ANSI_WARNINGS OFF
GO
```

```
ALTER DATABASE [EMCSQL64] SET ARITHABORT OFF
GO
```

```
ALTER DATABASE [EMCSQL64] SET AUTO_CLOSE OFF
GO
```

```
ALTER DATABASE [EMCSQL64] SET AUTO_CREATE_STATISTICS OFF  
GO
```

```
ALTER DATABASE [EMCSQL64] SET AUTO_SHRINK OFF  
GO
```

```
ALTER DATABASE [EMCSQL64] SET AUTO_UPDATE_STATISTICS ON  
GO
```

```
ALTER DATABASE [EMCSQL64] SET CURSOR_CLOSE_ON_COMMIT OFF  
GO
```

```
ALTER DATABASE [EMCSQL64] SET CURSOR_DEFAULT GLOBAL  
GO
```

```
ALTER DATABASE [EMCSQL64] SET CONCAT_NULL_YIELDS_NULL OFF  
GO
```

```
ALTER DATABASE [EMCSQL64] SET NUMERIC_ROUNDABORT OFF  
GO
```

```
ALTER DATABASE [EMCSQL64] SET QUOTED_IDENTIFIER OFF  
GO
```

```
ALTER DATABASE [EMCSQL64] SET RECURSIVE_TRIGGERS OFF
```

GO

ALTER DATABASE [EMCSQL64] SET DISABLE_BROKER

GO

ALTER DATABASE [EMCSQL64] SET AUTO_UPDATE_STATISTICS_ASYNC ON

GO

ALTER DATABASE [EMCSQL64] SET DATE_CORRELATION_OPTIMIZATION OFF

GO

ALTER DATABASE [EMCSQL64] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [EMCSQL64] SET ALLOW_SNAPSHOT_ISOLATION OFF

GO

ALTER DATABASE [EMCSQL64] SET PARAMETERIZATION FORCED

GO

ALTER DATABASE [EMCSQL64] SET READ_COMMITTED_SNAPSHOT OFF

GO

ALTER DATABASE [EMCSQL64] SET HONOR_BROKER_PRIORITY OFF

GO


```
ALTER DATABASE [EMCSQL64] SET READ_WRITE  
GO
```

```
ALTER DATABASE [EMCSQL64] SET RECOVERY SIMPLE  
GO
```

```
ALTER DATABASE [EMCSQL64] SET MULTI_USER  
GO
```

```
ALTER DATABASE [EMCSQL64] SET PAGE_VERIFY CHECKSUM  
GO
```

```
ALTER DATABASE [EMCSQL64] SET DB_CHAINING OFF  
GO
```

SQL Server 2008 changes after install:

- Database is set to simple recovery mode
- Max Degree of parallelism =4
- Mixed mode authentication
- Memory is capped so that O/s has 7 GB reserved. Cap 249 GB left to SQL Server, 7 GB reserved for O/s.
- -T1118 flag for tempdb contention
- Tempdb on its own LUN.
- 1 tempdb datafile for one CPU, 1 GB initial size.
- Lock pages in memory
- Forced parameterization
- Turned off Auto update statistics.
- For database, One file group with multiple files that will be spread across LUNs.
- Windows update has to be turned off.
- Statistics – Statistics for the database were completed using sp_updatestats command.
- We did not use the dm_updatestats job as this is outdated.