IBM® Content Manager V8.5 Enterprise Edition Performance Tuning Guide

SUPPORTING DB2 AND ORACLE ON AIX, WINDOWS, LINUX AND SOLARIS

Content Management Performance
IBM China Software Development Laboratory

BEIJING, CHINA P.R. 100193

September 2, 2013

TABLE OF CONTENTS:

About This Document

This publication is intended to help Content Management solution designers and system administrators understand the factors that impact the performance of IBM Content Management applications running on the supported platforms. This document provides guidelines for parameter and application tuning for IBM(R) DB2® Content Manager Enterprise Edition V8.5 for DB2 v9.x, v10.x and Oracle® 11g on AIX, Windows, and Linux platforms.

Performance tunings for the z/OS platform is not included in this paper.
Please visit the IBM(R) DB2(R) Content Management Enterprise Edition V8.5 information center http://publib.boulder.ibm.com/infocenter/cmgmt/v8r5m0/index.jsp for more information.

Tuning tips in this document apply to CM versions 8.5 GA unless otherwise indicated.

Parameter settings

This document provides guidance for configuring various performances related environmental settings. This guidance should be evaluated in context with the application workload and system configuration. In many cases, the suggested settings will need to be adjusted to optimize the performance of the product.

Assumptions

This document assumes you are a systems administrator and you have:

- Reviewed the *IBM CM 8.5 Hardware and Software Requirements* guide for your version of CM 8.5 and verified that CM 8.5 supports the hardware and software components in your current configuration.
- Installed CM 8.5 according to the *IBM CM 8.5 Platform Installation and Upgrade Guide* for your version of *IBM CM*.
- Configured the CM 8.5 databases with sufficient processing power and memory that are appropriate to the application.

Introduction

IBM® Content Manager Enterprise Edition V8.5 manages all types of digitized content including HTML and XML Web content, document images, electronic office documents, printed output, audio, and video.

The system diagram below shows the major components and tools in the Content Manager V8.5 architecture. The Content Manager system is scalable to fit the needs of small, medium, and large-scale businesses. The main system components that make up a Content Manager system are:

- Library Server
- Resource Manager
- Clients

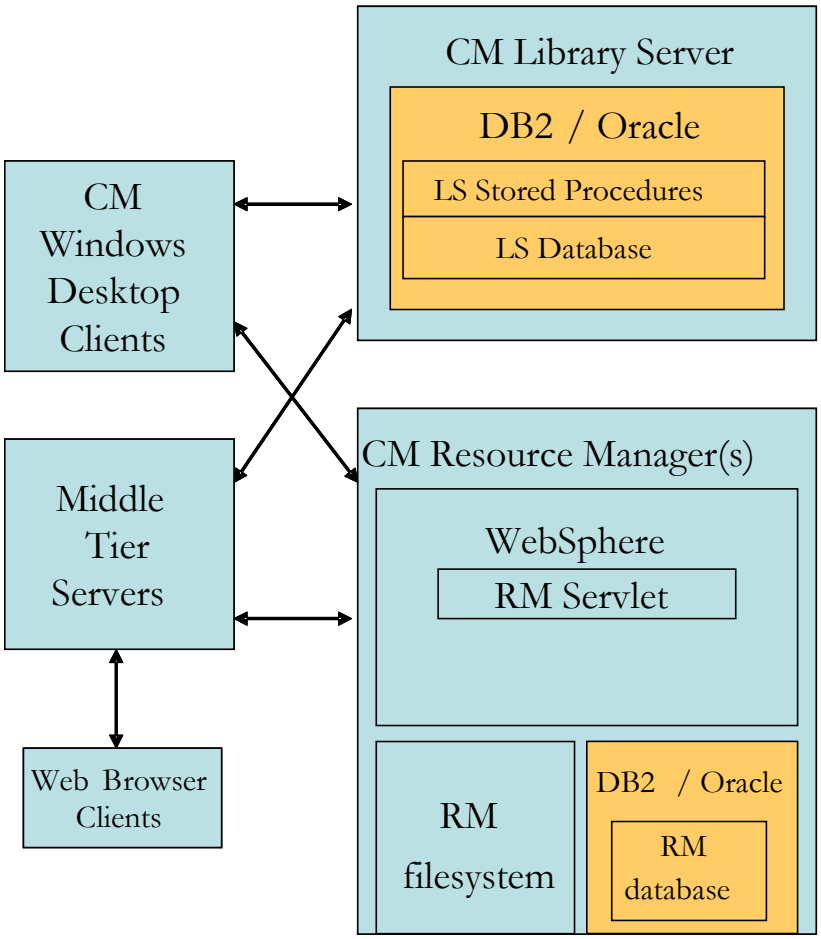


Figure 1. CM V8.5 High-level Architecture

The **Library Server** is the central source for indexing, describing, locating, organizing, and managing enterprise content. It supports the Content Manager data model, controls access to content, and manages transactions with referential integrity.

- Deep integration of the search engine with the underlying database in the Library Server supports search performance.
- A pre-built access control list on the Library Server supports user and user group privilege processing
- Closely associated objects can be assigned to a document;
- Documents can be grouped in one or more folders; and
- Folders can be placed inside of other folders.
- Users can assign indices to objects in the library to facilitate storage and retrieval. A search algorithm optimizes the time required to locate information. The system virtually eliminates loss and misplacement of file folders which is common in manual processing of information.

The **Resource Managers** are specialized repositories optimized to manage the storage, archiving, retrieval, and delivery of enterprise content. The Content Manager Resource Manager stores documents, images, and multimedia resources. It supports object management on many levels:

The **Mid-tier Servers** provide the Content Manager API toolkit and manages connections to the Library Server and to the Resource Managers. It is the Web-exploiting broker that:

- Mediates between the browser client and the Library Server.
- Leverages IBM® WebSphere® product families' connection pooling and load balancing capabilities

Content Manager Clients include the system administration client, Client for Windows, WEBi and IBM Content Navigator. The clients running either on user desktops or in mid-tier application servers invoke Content Manager Services that are divided between a Library Server and one or more Resource Managers.

CM Performance Methodology

The CM Performance Methodology has a five-stage lifecycle. By setting objectives early, performance tuning before going into production, and by establishing processes that monitor key performance metrics, the methodology will help ensure achievement of initial scalability objectives and will also provide information for capacity planning as workloads change and grow.

Planning for performance and scalability
Document the projected system topology, the projected workload, and define measurable performance and scalability objectives. Perform initial capacity planning and sizing estimates.

Solution design choices and performance tradeoffs
Understand topology choices (types of IBM Content Navigator, Client for Windows, Custom client), feature choices (e.g. replication, High Availability, text search, versioning, etc) and their performance implications.

Performance tuning
Plan for an initial tuning period to maximize confidence and reduce risk, using the tools and techniques from the Monitoring and Troubleshooting phases to meet the objectives set in the planning phase. Focus on effective tuning of CM server calls, memory, disk I/O, and network bandwidth. The sections that follow provide key parameters in various CM components that will give you the greatest performance impact.

Monitoring and maintaining performance
Maintain a .performance profile of the workload metrics and resource utilizations on your CM servers. Monitor over time to observe trends before they become a problem.

Performance troubleshooting
When performance issues arise, use a disciplined and organized approach to solve the problem using the performance profile data and performance tuning guidelines.

Performance Tuning Best Practices

Performance problems generally result from a lack of throughput or unacceptable response times which are either sporadic or regular. The problem might be localized to specific application modules, or it might span the entire system.

Before starting the performance tuning cycle, you first do some preparatory work that establishes the framework for ongoing performance tuning activities. You should:

- Identify constraints — The business case determines priorities, which in turn establish boundaries. Constraints, such as maintainability and budget limits, are unalterable factors in search of higher performance. You must focus performance work on factors that are not constrained.
- Specify the load — Involves determining what services the clients connecting to the Library Server or Resource Manager require and the level of demand for those services. The most common metrics for specifying load are the number of clients, client think time (the amount of time between the receipt of a reply by a client and the subsequent submission of a new request), and load distribution (steady or fluctuating, average, and peak load).
- Set performance goals — Performance goals must be explicit, which involves identifying the metrics used for tuning as well as their corresponding benchmark values. Total system throughput and response time are two common metrics used to measure performance. After identifying the performance metrics, you must establish quantifiable and reasonable benchmark values for each one.

After establishing the boundaries and expectations for performance tuning, you can begin the tuning cycle, which is an iterative series of controlled performance experiments. Removing the first bottleneck might not lead to performance improvement immediately, because another bottleneck might be revealed that has an even greater performance impact on the system.

Identify constraints

IBM® Content Manager Enterprise Edition V8.5 sets many production-ready performance tuning parameters right out of the box, especially for the Library Server. But additional tuning will often be required to maximize the performance and scalability of a production system, and you should plan for an initial tuning period to maximize confidence and reduce risk. Performance tuning for an IBM Content Manager system depends on coordinated and interdependent tuning of the operating system, databases, the system memory usage, disk I/O activity, CPU usage, and network usage. Library Server, Resource Manager and middle-tier servers may each have different tuning specifications. Understanding the way the Application uses & manages the system resources helps us understand what subsystems we need to tune in any application scenario. Always remember that that performance tuning involves tradeoffs; appropriate tuning techniques and parameter values depend on the unique circumstances of your configuration and workload.

Performance tuning requires a different method to the initial configuration of a system. Configuring a system involves allocating system resources in an ordered manner so that the initial system configuration is functional.

Tuning is driven by identifying the most significant bottleneck and making the appropriate changes to reduce or eliminate the effect of that bottleneck. Usually, tuning is performed methodically, measuring the effect of each tuning parameter change on resulting system throughput or response times.

Sizing a Content Manager system

To start the Tuning exercise, we need to understand how the existing system is dealing with its content. This includes:

- Document types and counts

Documents should be grouped and categorized by type (for example, invoices or claims). These groupings will be based on the way the documents are to be treated for security, and indexing data and storage requirements. The groups will probably translate to Content Manager item types.

- Document arrival rate

When, where, and how many documents are entering the system?
Although annual document volumes are often used as a benchmark, it is dangerous to apply these statistics without knowing document arrival peaks and valleys. It is critical to identify the peak arrival rate and size the system accordingly. It is also useful to understand the locations and formats of the documents that can be captured. Are all application forms sent to a central mail room and processed there, or are they sent to individuals in branch offices?

- Document usage

When, where, and how is each kind of document used?
For example, one type of form might go through a business process, following a specific workflow. The workflow process comprises seven different activities by different people. At each workflow step, this document is viewed by the person performing the activity. This example means that each form is retrieved at least seven times. That does not seem like much, but if there are 10,000 of these forms arriving each day, then there are 70,000 retrievals happening each day.

Set performance goals

What is the measure of acceptable performance?
How many transactions an hour, or seconds of response time will meet the required performance level?

Performance goals can be divided into three major categories:

- Response-time goals
  Goals are set for an "acceptable" response time for a single operation. Example: how long a database query takes and how long it takes to display a retrieved document.
- Throughput goals
  Given "acceptable" response time for a set of operations, it is the number of workload operations that can be accomplished per unit of time. Example: Library server database transactions or stored procedures per minute.
- Scalability goals
  Given maximum transaction throughput, determine how many concurrent users can be served by the system. Goals are set to achieve, for example, 25000 views/hr with 400 concurrent users on a system.

Almost every client action interacts with the server and causes some load generated on that server. From logon through search, retrieve, update, and delete, they all generate work for the server.

Stored Procedure Calls (SPCs) and HTTP requests can be used as the workload metrics. Stored Procedure Calls represents a generic unit of work for the Library Server and the HTTP request represents a unit of work for the Resource Manager.

To determine your workload, you can trace a sample workload. To measure a workload, use the Library Server performance-level trace to monitor a sample workload on a test system client, count the number of SPCs generated, and multiply that by the total number of requests determined by the questions above.

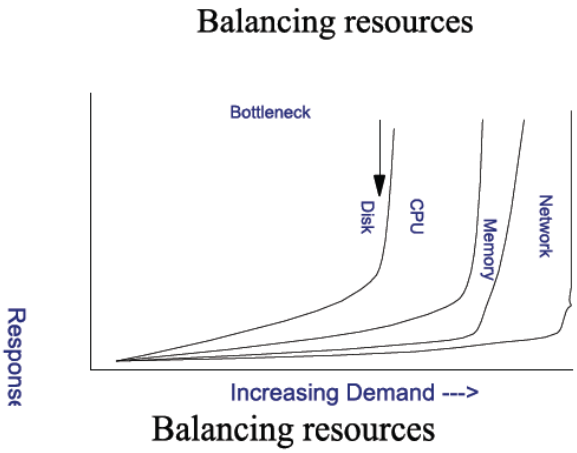
This will give an estimate of the number of SPCs likely to be generated by the system.

The Resource Manager is an application that uses processor and memory capacity as well as I/O capacity for growth. Usually, the largest storage usage in the system is the Resource Manager file system or LBOSDATA directory. To estimate the required amount of storage, simply multiply the average file size by the number of files expected to be stored here. The amount stored here will continue to grow while new data is added and old data is maintained. Old data may be removed from the system by deliberate deletions after the data is no longer required, or because it is migrated to another storage system using the Tivoli Storage Manager(TSM).

Equipped with this information, we can identify potential bottlenecks in physical or logical resource performance.

Stack model for Performance Tuning

In the CM system, when performing a task for any resource, a minimum amount of that resource is needed to accomplish the task. A bottleneck is any part of the CM system spending resources for poor reasons and spending more than necessary.



It is important to realize that during tuning you are minimizing the impact of bottlenecks, not eliminating them. Tuning allows you to adjust resources to achieve your CM application performance objectives. For the scope of this document, this includes:

- Tuning the CM application
- Tuning the Operating System
- Tuning Storage parameters & I/O layout
- Tuning the DBMS
- Tuning WebSphere Server Performance Parameters & JVM
- Tuning Tivoli storage (optional – only if using Tivoli for migration)

The tasks above represent a road map for performance tuning of the Content Manager system. The tasks presented here are the most common suggestions for tuning an initial install.

New performance related features in CM 8.5

Release	Feature
8.5 GA	Library Server stored procedures (SPs) enabled SQL parameter marks for performance improvement. Support for the following new levels of key prerequisite software: • DB2 V10.5 • WebSphere Application Server V8.5 • Java™ 7.0

Tuning the CM application

As you plan your Content Management system, wizards can guide you through installing, configuring, upgrading, or removing IBM® Content Manager. The steps below help guide the user towards options during the initial planning, choice of Hardware options, software installation & configuration stages that would help the performance of the Content Management system.

Stage	Tool / Option	Benefits/ Description
Planning & Platform choices	Choice of Platform CPU	Choose 64-bit processors with large L2 or L3 processor caches because of the benefit of additional address space & speed. It is important to match and scale the memory and I/O subsystem with the CPU power and vice versa.
	Memory (RAM), Swap and Paging Storage	Increase the RAM to match your memory needs. As a rule-of-thumb allocates the Library Server at least 4x times the memory compared to the Resource Manager. For Swap partitions, create a partition sized at 2x RAM on the fastest disks available. Place the page file and operating system files on separate physical disk drives. Use multiple disks or a disk array if additional disk bandwidth is needed for paging. Do not place multiple page files on different partitions of the same physical disk drive.
	Library server platform choice	You can only install one library server per server per CM system. Library server systems have high reading and writing work load requirements, they require a powerful processor to accommodate concurrent requests from multiple users. Choose the highest number of CPU's & memory for the Library system that you can afford.
	Resource manager platform choice	The resource manager is the repository for objects stored in the system. While the total # of CPU's could be lower than the Library server, the storage capacity & speed is critical on these systems. Choose the fastest & largest storage for this system. The faster storage should be allocated to database redo logs & staging. The large size storage capacity is to be used for /lbosdata, the default directory for object storage. The Resource managers can be distributed across networks to provide convenient user access & horizontal system scalability. Place resource managers in several locations to enable users to access servers that are closer to them.
	System Topology options	Install the library server, resource manager, system administration client, and mid-tier servers if applicable, on separate servers to increase performance and make the system more efficient.
	Topology of resource managers	Multiple resource managers give higher total bandwidth for larger objects. Multiple resource managers give higher migrator parallelism.

	Choice of Platform OS	We recommend using x64 bit platforms v/s x32 bit platforms to take advantage of larger address space & achieve scalability
	"sizer" tool	Your IBM® representative has a "sizer" tool to help you make an initial rough sizing of the hardware configurations that should be able to support your workload.

Stage	Tool / Option	Benefits/ Description
Installation	System requirements – hardware and software	http://publib.boulder.ibm.com/infocenter/cmgmt/v8r5m0/index.jsp

Stage	Tool / Option	Benefits/ Description
Configuration	switch IBM® Content Manager to 64-bit for use with 64-bit DB2	For AIX®, Solaris SPARC, and Linux® x86, starting at V8.4.2 you can switch from 32-bit to 64-bit storae procedures by configuring for 64-bit using the 'cmcfglsi –t to64' command.
	switch user exits from 32-bit to 64-bit	For AIX®, Solaris SPARC, and Linux® x86, starting at V8.4.2 you can switch your user exits from 32-bit to 64-bit by recompiling them. Your DB2 Content Manager library server must have been configured for 64-bit using the cmcfglsi –t to64 command.
	Database files, logs, Staging & lbosdata storage.	Allocate dedicated drives for the following: Library Server database Library Server database indexes Library Server database log files Resource Manager database Resource Manager database indexes Resource Manager log files Resource Manager LBOSDATA directory Resource Manager Staging directory Backup directories Install the Resource manager staging area and lbosdata on dedicated drives or file systems. If the staging area and lbosdata area are not on dedicated drives or file systems, the Resource Manager purger and threshold migration may not function correctly & could affect performance.

Operating System Tuning

This section describes tuning options within operating systems supported by the CM V8.5 application.

AIX

The objective for performance tuning the AIX Operating system is to optimize the available system resources. AIX systems can be tuned for CPU, memory, disk, and network performance.

On POWER7-series servers and virtualization environments, the default FIPS-certified version of IBM Global Security Kit (GSKit) might cause performance degradation. The non-FIPS certified version of GSKit contains a performance fix by setting the ICC_IGNORE_FIPS and ICC_TRNG environment variables. See below link for more details.

http://www-01.ibm.com/support/knowledgecenter/SSRS7Z_8.5.0/com.ibm.installingcm.doc/dcmit371.htm

CPU tuning guidelines

Tuning parameter	Description/ Function
Prioritizing CM related processes	By default, the CM related processes have a base priority of 60 and a nice value of 20. Consider changing the priority of a CM process or thread. For the Library Server installed on DB2, the main CM related processes are the db2sysc and db2fmp processes. On Oracle, the main CM related processes are the extproc & oracle processes. For the Resource Manager server, the main CM related processes are the Java and db2sysc processes for DB2. For Oracle they are Java & oracle processes.

Memory tuning

Parameter Type	Description/ Function
Controlling Buffer-Cache Paging Activity	Excessive paging activity decreases performance substantially. This can become a problem with database files created on AIX journaled file systems (JFS and JFS2). The behavior of the AIX file buffer cache manager can have a impact on performance. It can cause an I/O bottleneck, resulting in lower overall system throughput.
Vmo tunables	MINPERM : The minimum number of permanent buffer pages for file I/O.
	MAXPERM : The maximum number of permanent buffer pages for file I/O.
	MINFREE: The minimum free-list size. If the free-list space in the buffer falls below this size, the system uses page stealing to replenish the free list.
	MAXFREE: The maximum free-list size. If the free-list space in the buffer exceeds this size, the system stops using page stealing to replenish the free list.

	The default values of the **minfree** and **maxfree** parameters depend on the memory size of the machine. The difference between the **maxfree** and **minfree** parameters should always be equal to or greater than the value of the maxpgahead parameter, if you are using JFS. For Enhanced JFS, the difference between the **maxfree** and **minfree** parameters should always be equal to or greater than the value of the j2_maxPageReadAhead parameter. If you are using both JFS and Enhanced JFS, you should set the value of the **minfree** parameter to a number that is greater than or equal to the larger pageahead value of the two file systems.
	MAXCLIENT: represents the maximum number of client pages that can be used for buffer cache by enhanced JFS file systems if the strict_**maxclient** parameter is set to 1, which is the default value.
	LRU_FILE_REPAGE: The default setting for lru_file_repage is 1 (on). When the **lru_file_repage** is set to 1 (the default), then the file and computational repage rates are considered when deciding what type of page to steal. If **lru_file_repage** is set to 0, then the repage rates are ignored and the page stealer will try to only steal from file pages, as long as *numperm* is greater the *minperm*. You should tune it according to your system profile and needs.

Storage tuning guidelines

Storage tuning involves the **physical** components (controllers/adapters, drivers, disks, cache, # of spindles etc), **logical** components (LVM, file systems, RAID) & **layout** goals including:

- Minimize I/O service times
- Balance I/O across all disks
- Keep I/O localized and sequential

For more detailed tuning, please refer to AIX storage Performance Tuning Guide at :
http://www.ibm.com/developerworks/aix/library/au-aixoptimization-disktun3/index.html

Network tuning guidelines

Tuning parameter	Description/ Function
Adapter speed and duplex mode settings	Set the NIC & switch to Auto_Negotiation mode which negotiates the speed and duplex settings for the highest possible data rates.
MTU setting	Properly configure all the NIC, devices and switches to the largest common MTU that all devices support. Large MTU sizes allow the operating system to send fewer packets of a larger size to achieve the same network throughput.

On Gigabit Ethernet networks	Select the jumbo frame mode. Ensure that the entire network including all networking devices on the subnet are using the same payload size.
TCP Window size	Enable the RFC1323 option. TCP's window size can be set as high as 4 GB.
TCP socket buffers – Sending tcp_nodelay	On the sending side, the packet can sit at the socket until the amount can fill the send buffer or be at least as large as the maximum segment size or until the fasttimeout timer has expired. The socket can be set to tcp_nodelay to allow the sender to transmit data right away without waiting. This can also improve performance of request/response type of applications. We recommend you to use tcp_nodelay feature and send immediate acknowledgement packets to improve network performance, especially for secure transport enabled.
S/W Transmit Queue Overflow	It indicates the number of outgoing packets which have overflowed the transmit queue. If there is a non-zero value for the field, you should increase the transmit queue size for the adapter.
TCP receive buffer	The IP layer has an input queue. The Demux layer places incoming packets on this input queue. If the queue is full, packets are dropped and never reach the application. If the statistic ***ipintrq overflows*** increases in value, then you should increase the **ipqmaxlen**.
Ultimate limit for TCP and UDP buffers	The **sb_max** controls the upper limit for any socket buffer.

Other System Resources

Tuning parameter	Description/ Function
maxuproc	Since each user is handled by each DB2 **db2fmp** process in Content Manager, you need to adjust the maximum number of PROCESSES allowed per user on the database instance owner depending on how many concurrent users will be connected to your system. For a production system with either database, that handles a high number of concurrent users, it is recommended to set this parameter to at least 4096.

ulimit	File descriptor specifies the limit on the number of open files a process may have. It sets and reports user **process resource limit**, as defined in the /etc/security/limits file.
	
	• We recommend setting the file size and size of the data area to unlimited by editing the /etc/security/limits file for both Library Server and Resource Manager.
	• Under the db2 instance owner (db2inst1), add the following values:
	  o fsize = -1
	  o data = -1

Windows 2008 Enterprise Server

Windows Server® 2008 should perform well out of the box for most Content Manager workloads. This section describes important tuning parameters and settings that can result in improved performance for CM on the Windows Server® 2008 operating system.

CPU and Memory tuning guidelines

Feature	Description & tuning recommendation
Affinitize application threads to a core	Windows 2008 Interrupt affinity feature allows the user to bind interrupts from a specific device to specific processors in a SMP system. The binding forces interrupt processing to run on the specified processor or processors. It helps improve system locality and scalability of the CM application. Use the Interrupt-Affinity Policy (IntPolicy) tool for systems with more than eight logical processors or for devices that use MSI or MSI-X.
Disable unnecessary Windows services	When system is running, many services are automatically enabled that may not be necessary for a particular server. In our tests, we found CPU usage dropped dramatically by disabling non essential Windows services.
Enable Hyper-Threading	Hyper-Threading technology on Intel based CPU's enable multi-threaded software applications to execute threads in parallel. A processor with hyper-threading enabled is treated by the operating system as two processors instead of one and shares the workload between them. CM applications support multi-threaded code, allowing multiple threads to run simultaneously yielding improved reaction and response time.
High Performance Power plan	The default power plan for Windows Server 2008 is "Balanced". This option is optimized for maximum power efficiency matching computational capacity to computational demand by dynamically reducing or increasing CPU performance as workload changes. The default plan keeps performance high and saves power whenever possible. Change the power plan to High Performance to achieve better performance. Ensure that Server BIOS settings are set appropriately such that Windows 2008 power plans will operate as designed.

Storage tuning guidelines

Windows 2008 has a layered storage stack model. Here are the components of the stack that are traversed by an application call and each component could make an impact on the overall I/O performance.

Layer	Components & Options	Description
File System	NTFS	We recommend that you use NTFS because of its advanced performance, security, and reliability features. Several factors affect NTFS Performance including Cluster Size, location and fragmentation of Master File Table (MFT) paging file, and NTFS Volume compression. Master File Table (MFT) fragmentation: Starting with Windows Server 2008, the MFT uses a fixed 200MB zone reservation so as not to reserve space that will never actually be used. Adjust the size the MFT zone reservation uses with a valid range of 1-4 where this is a multiplier of 200MB. Choose a higher value if your volume sizes are going to be large. Disable NTFS file compression. Disable the 8.3 file name creation on the Resource Manager to avoid the NTFS overhead of creating the short name files for Content Manager long-name (30 CHARS) files that are stored in the LBOSDATA area
Volume & Management	Volume Manager, VolSnap	The Windows volume manager stripe unit is fixed at 64 KB. Hardware solutions can range from 4 KB to 1 MB and even more. Ideal stripe unit size maximizes the disk activity without unnecessarily breaking up requests by requiring multiple disks to service a single request. Place different types of workloads into separate volumes on different virtual disks. Place the paging file on a separate virtual disk to provide improvements in performance during periods of high paging. The "first" partition on a volume usually uses the outermost tracks of the underlying disks and therefore provides better performance.
Partition & Class	PartMgr, Disk	For maximum performance and reliability, the storage controllers should offer RAID capabilities. To determine the optimal RAID level, evaluate the read and write loads of all transaction types and then decide how much you can spend to achieve the performance and availability/reliability that your CM deployment requires
I/O Adapter Interface	Mini Port	NumberOfRequests : A higher value might improve performance and enable Windows to give more disk requests to a logical disk, which is most useful for hardware RAID adapters that have concurrency capabilities. I/O Priorities: Windows Server 2008 can specify an internal priority level on individual I/Os. Windows primarily uses this ability to de-prioritize background I/O activity and to give precedence to response-sensitive I/Os.
Disk Interface	SAS or SATA or	Power Protection and Advanced Performance Option when write

SCSI	caching is enabled; for every disk • Enable write caching. • Enable an "advanced" performance mode that assumes that the storage is protected against power failures.

Network tuning guidelines

The following features supported by Windows 2008 enable use of the network interface features. For Windows servers, use Windows registry to update the TCP layer parameters.

Registry Entry	Parameter and Value
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TCPIP\Parameters	TcpTimedWaitDelay= 30 (decimal) MaxUserPort= 32768 (dec) KeepAliveInterval= 1 (dec)
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\	TcpAckFrequency= 1
Tcpip\Parameters\Interfaces\ID	TcpMaxDataretranmissions= 5 (dec)
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\AFD\Parameters	"EnableDynamicBacklog"=dword:00000001 "MinimumDynamicBacklog"=dword:00000020 "MaximumDynamicBacklog"=dword:00001000 "DynamicBacklogGrowthDelta"=dword:00000010

Linux

This section covers optimizations for Linux x86_64 platforms We will not be covering 32 bit OS versions as the advantages & feature set with 64 bit OS's are numerous and hence is the suggested choice for deployment.

Tuning the Linux Kernel

Linux distributions and the Linux kernel offer a variety of parameters and settings to let the Linux administrator tune the system to maximize performance.

Resource	Tool usage	Tune
CPU	ps –ef	Schedule non prioritized processes to run at off-peak hours.
	Renice	Identify non-critical, CPU-intensive processes and modify their priority.
	Taskset	Bind processes to CPUs to make sure that processes are not hopping between processors, causing cache flushes.
Kernel type	SMP	The Kernel has support for SMP and hyper-threaded machines. Some packages also include support for NUMA. There may be some variant, depending on the amount of memory, the number of CPU, and so on.
	X64 bit Linux kernel	On 32-bit architectures, the maximum address space that single process can access is 4GB. On the other hand, with 64-bit architectures such as x86-64 (also x64), it extends all the way to 64 GB or to 128 GB in the case of IA-64 systems. We strongly recommend using x64 bit kernels.
File system selection and tuning	journaling file systems	Choose a journaling file system as part of the default installation. (ext3, jfs, reiser) Journaling also improves recovery time, because there is no need to perform file system checks at system reboot. Generally speaking, ReiserFS is more suited to accommodate small I/O requests whereas XFS and JFS are tailored toward very large file systems and very large I/O sizes. Ext3 fits the gap between ReiserFS and JFS/XFS since it can accommodate small I/O requests while offering good multiprocessor scalability. Our tests comparing ReiserFS & ext3 file systems showed little difference in CM performance.
Runlevel selection	who –r OR runlevel	With the GUI disabled locally on the server, you can connect remotely and use the GUI. To do this, use the -X parameter with the ssh command.
	telinit [-t sec] <Run Level Number> OR "init 3"	We suggest using runlevel 3 for all server systems.
Resources available to the shell	ulimit –a	The -H and -S options specify the hard and soft limits that can be set for the given resource. If the soft limit is passed, the system administrator will receive a warning.

		The hard limit is the maximum value that can be reached before the user gets the error messages Out of file handles.

	/etc/security/limits.conf	To limit users on the fly:
		
		`userid soft nproc 2047`
		`userid hard nproc 16384`
		`userid soft nofile 4096`
		`userid hard nofile 65536`
		
		In addition, make sure that the default pam configuration file (/etc/pam.d/system-auth for Red Hat Enterprise Linux, /etc/pam.d/common-session for SUSE Linux Enterprise Server) has the following entry:
		
		session required pam_limits.so
		
		This entry is required so that the system can enforce these limits. For Oracle deployments the userid could be "oracle".

disk subsystem	number of disk drives & partitions	With RAID technology, you must spread the I/O over multiple spindles & controllers. When the OS cache and the cache of the disk subsystem can no longer accommodate the amount or size of a read or write request, the physical disk spindles start to work. During the installation process, choose to create a multi-partition layout.
	block size	Benchmarks have shown there is negligible performance improvement by changing the block size of a file system. Leave it at the default of 4 K.
	RAID stripe-unit size	The *stripe-unit size* is the granularity at which data is stored on one drive of the array before subsequent data is stored on the next drive of the array. Selecting the correct stripe size is a matter of understanding the predominant request size performed by the CM application. The stripe size of a hardware array has, in contrast to the block size of the file system, a significant influence on the overall disk performance.
Quotas	Session limits	Ensure that there are **no** limits set for any CM userid's such as icmadmin, including database users & root for resources listed below. They are typically configured in the /etc/security/limits.conf file. memlock - Maximum locked-in memory address space (KB). rss - Maximum resident set size (KB). stack - Maximum stack size (KB). cpu - Maximum CPU time in minutes. as - Address space limit. maxlogins - Maximum number of logins allowed for this user.

Modifying kernel parameters: Oracle 11g

Verify that the kernel parameters shown in the Oracle Doc ID 169706.1 are set to values **greater than** or equal to the recommended value shown for an Oracle 11g installation. If the current value for any parameter is higher than the value listed in this table, do not change the value of that parameter.

Modifying kernel parameters: DB2 9.7 & DB2 10.x

The default values for particular kernel parameters on Linux are not sufficient when running a DB2 database system. Before installing a DB2® database system, you should update your Linux® kernel parameters.

Refer to DB2 solution information center at:
http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/index.jsp
http://publib.boulder.ibm.com/infocenter/db2luw/v10r1/index.jsp
http://publib.boulder.ibm.com/infocenter/db2luw/v10r5/index.jsp

- Look for: **Related tasks (**Modifying kernel parameters (Linux)

Network tuning guidelines

Resource	Tool usage	Tune
Network subsystem	Speed and duplexing ethtool command	Check the actual speed of the network interface, because there can be issues between network components (such as switches or hubs) and the network interface cards. The mismatch can have a large performance impact.
buffer sizes	Tuning TCP for Linux 2.4 and 2.6	Since autotuning and large default buffer sizes were released progressively over a succession of different kernel versions, it is best to inspect and only adjust the tuning as needed.
increase TCP throughput	Increase the size of the interface queue for Gigabit Ethernet connected hosts.	ifconfig eth0 txqueuelen 1000
	TCP algorithm	Verify that the following are all set to the default value of 1 : sysctl net.ipv4.tcp_window_scaling sysctl net.ipv4.tcp_timestamps sysctl net.ipv4.tcp_sack
	Autotuning TCP buffer limits	net.ipv4.tcp_rmem = 4096 87380 16777216 net.ipv4.tcp_wmem = 4096 65536 16777216

	TCP buffer size	net.core.rmem_max = 16777216 net.core.wmem_max = 16777216

Memory tuning guidelines

In Linux, the size of main memory is often more important than the processor speed. One reason, if not the main reason, for this is the ability of Linux to create dynamic buffers containing hard disk data. For this purpose, Linux uses various techniques, such as read ahead (reading of sectors in advance) and delayed write (postponement and bundling of write access).

Resource	Tool usage	Tune
Memory	sar –B	Provides useful information for analyzing page faults, specifically columns pgpgin/s and pgpgout/s.
		Hard page faults occur when the page is not found in memory and must be fetched from disk.
	SHMMAX	A shared memory parameter SHMMAX is the maximum size in bytes, of a shared memory segment. Allocate this to half the size of physical memory (in bytes)
	SHMALL	This parameter sets the total amount of shared memory pages that can be used system wide. Hence, SHMALL should always be at least ceil(shmmax/PAGE_SIZE).
	Free	Swap space usage.
		If you see constant swapping you need to either add more RAM or reduce the size of the database memory allocation. Constant swapping should be avoided at all cost.
Swap file system	use a swap partition	Use a swap partition if possible. Swap partitions, as opposed to swap files, have a performance benefit because there is no overhead of a file system.
		Size of the swap partition should be two times physical RAM. Create multiple swap partitions. Linux can take advantage of multiple swap partitions or files and perform the reads and writes in parallel to the disks.

		Swap partitions are used from the highest priority to the lowest (where 32767 is the highest and 0 is the lowest). Giving the same priority to the partitions causes data to be written to the partitions in parallel and performance generally improves.
	swappiness	By default, Linux will aggressively swap processes out of physical memory onto disk in order to keep the disk cache as large as possible. This means that pages that haven't been used recently will be pushed into swap long before the system even comes close to running out of memory, which is an unexpected behavior compared to some operating systems. The /proc/sys/vm/swappiness parameter controls how aggressive Linux is in this area. The optimal setting here is dependant on workload. In general, high values maximize throughput: how many transactions the system gets down during a unit of time. Using 100 will always favor making the disk cache bigger and yielded good results during tests in the IBM CM Performance Labs.

The following table describes the relationship between installed RAM and the configured swap space requirement.

RAM	Swap Space
Up to 512 MB	2 times the size of RAM
Between 1024 MB and 2048 MB	1.5 times the size of RAM
Between 2049 MB and 8192 MB	Equal to the size of RAM
More than 8192 MB	0.75 times the size of RAM

Solaris 10

The Solaris OS is a multi-threaded, scalable UNIX® operating system that runs on SPARC and x86 processors. It is self-adjusting to system load and demands minimal tuning. In some cases, however, tuning is necessary. The Solaris 10 OS is proven to support 144+ processing units and has features such as Dynamic Tracing (DTrace), Solaris Zones, and Resource Pools that enable administrators to configure the Solaris platform for the application workload to run well on the underlying server system.

In Solaris 10, the System V IPC parameters have been made dynamic. That is, a parameter value can be changed on-the-fly using the new Solaris 10 resource control facilities. Also, many of the older parameters are obsolete, and remaining parameters have much larger default values.

Tuning the Solaris Kernel

In previous Solaris releases, IPC facilities were controlled by kernel tunables. Since the IPC facilities are now controlled by resource controls, their configuration can be modified while the system is running.

Many applications that previously required system tuning to function might now run without tuning because of increased defaults and the automatic allocation of resources. If /etc/system entries exist in Solaris 10, and the set values are larger than the new default values, the settings will work system-wide.

Parameters type	Tunable parameter	Description	Range
General Kernel and Memory	Physmem	Number of usable pages of physical memory available on the system.	physical memory on system
Process-sizing	Maxusers	Users	1 to 4096, if set in the /etc/system file
	max_nprocs	Maximum number of processes that can be created on a system.	10 + (16 x maxusers)
	maxuprc	Maximum number of processes that can be created on a system by any one user.	max_nprocs - reserved_procs
Swapping	swapfs_reserve	Amount of system memory that is reserved for use by system (UID = 0) processes.	The minimum value is 4 Mbytes or 1/16th of physical memory, whichever is smaller, expressed as pages using the page size returned by getpagesize.
	swapfs_minfree	Desired amount of physical memory to be kept free for the rest of the system.	The larger of 2 Mbytes and 1/8th of physical memory
Paging	lotsfree	When this threshold is crossed, the page scanner wakes up to begin looking for memory pages to reclaim.	maximum value should be no more than 30 % of physical memory, in units of pages
	desfree	Preferred amount of memory to be free at all times	lotsfree / 2

	maxpgio	Maximum number of page I/O requests that can be queued by the paging system.	1 to a variable maximum that depends on the system architecture, but mainly by the I/O subsystem, such as the number of controllers, disks, and disk swap size
File descriptors	ulimit	Maximum number of open files supported.	If the value of this parameter is too low, a *Too many files open* error is displayed in the WebSphere Application Server stderr.log file.

Memory tuning guidelines

The following parameters are important when configuring and tuning Caché on the SUN Solaris version platform:

The NOFILES parameter is unlimited. Edit the /etc/system file and add the following variables to increase shared memory segments:

Variable	Description
shmsys:shminfo_shmmax	Maximum shared memory segment size
shmsys:shminfo_shmmin	Minimum shared memory segment size
shmsys:shminfo_shmmni	Number of shared memory identifiers
shmsys:shminfo_shmseg	Number of segments, per process
semsys:seminfo_semmap	Number of entries in the semaphore map
semsys:seminfo_semmni	Number of semaphore identifiers
semsys:seminfo_semmns	Number of semaphores in the system
semsys:seminfo_semmsl	Maximum number of semaphores, per ID
semsys:seminfo_semmnu	Number of processes using the undo facility
semsys:seminfo_semume	Maximum number of undo structures per process

Oracle® Specific Kernel Settings

On Solaris 10, verify that the kernel parameters shown in Oracle Doc ID 169706.1 are set to values greater than or equal to the recommended value shown. For production database systems, Oracle® recommends that you tune these values to optimize the performance of the system.

SWAP Tuning Guidelines

Developing a memory sizing methodology requires resources to be broken into a system wide category and a per-process category. The goal of sizing the memory requirements of a system or

application is to minimize paging. On a server system, it is usually possible to eliminate almost all paging by configuring enough memory to run the required applications.

The following table describes the relationship between installed RAM and the configured swap space requirement.

Physical RAM	Swap Space
Between 1024 MB and 2048 MB	1.5 times the size of RAM
Between 2049 MB and 8192 MB	Equal to the size of RAM
More than 8192 MB	0.75 times the size of RAM

Network tuning guidelines

It is important to make sure the physical network is as fast as possible. Use a gigabit speed network switch to connect multiple servers & clients. A controller that is consistently overloaded or shows a high number of collisions is a potential bottleneck. If the NIC is overloaded, consider configuring a second controller.

The following table summarizes some tunable options:

Parameter	Description & Tuning
1000 Base-T NIC	Set the network switch at 1000 Mbps and full duplex; the server NIC with auto-negotiation enabled when connected to that switch will attain the fastest speed.
Maximum Transmission Unit (MTU)	1000Base-T supports Jumbo Frames only at 1000 Mbps with an MTU of 9000 bytes. Regular 10/100 Ethernet MTU is (1500 bytes).
If using 10/100 Mbps NIC	Disable auto-negotiation for Ethernet cards and set them to the fixed media speed and duplex mode. Set the same setting at the switch port.
RFC1323 timestamps, window scaling and RFC2018 SACK	RFC1323 timestamps, window scaling and RFC2018 SACK should be enabled by default. ndd /dev/tcp tcp_wscale_always #(should be 1) ndd /dev/tcp tcp_tstamp_if_wscale #(should be 1) ndd /dev/tcp tcp_sack_permitted #(should be 2)
Buffer size	Set the maximum (send or receive) TCP buffer size the CM application can request ndd -set /dev/tcp tcp_max_buf 4000000
Maximum congestion window	ndd -set /dev/tcp tcp_cwnd_max 4000000
Default send and receive buffer sizes	ndd -set /dev/tcp tcp_xmit_hiwat 4000000 ndd -set /dev/tcp tcp_recv_hiwat 4000000

Storage tuning guidelines

In Solaris, storage is managed by volume managers and file systems - the volume manager manages the lower level aggregation of disks into virtual volumes, and the file system provides the Unix administration model based on files.

Volume Management:

While Solaris Volume manager provides effective volume management, we recommend Hardware volume management since it unloads the host of the CPU overhead of managing RAID, and provides efficient RAID-5 implementations.

We recommend the use of striping and/or RAID-5 to aggregate storage, and eliminate the need for individual disk management

UFS File System

UFS is the default disk-based file system in Solaris operating environment which we recommend using. UFS provides the following tuning options:

UFS Feature	Description & Tuning
Direct I/O	An efficient way for Oracle® and DB2® to perform I/O is to have the operating system bypass it's file system cache, and read blocks from disk straight into database buffer cache. This has the advantage of retaining the regular file system administration model, with the advantage of providing an efficient code path similar to that of raw disk. The Direct I/O feature is enabled by mounting the file system with the forcedirectio option. It provides a substantial performance improvement for three reasons: • It eliminates double buffering (at database & OS) • It provides a small, efficient code path for reads and writes • It removes memory pressure from the operating environment
Asynchronous I/O	Asynchronous I/O enables a single application thread to overlap I/O operations with other processing, by providing an interface for submitting one or more I/O requests in one system call without waiting for completion. It enables I/O concurrency for greater performance on multidisc storage. The asynchronous I/O implementation is enabled by setting the Oracle® init.ora parameter disk_async_io. Concurrent writes are handled by operating system threads, allowing a single Oracle® database writer process to overlap up to 3072 asynchronous writes. We observed better performance using Asynchronous I/O in lab tests.

The Solaris operating environment automatically creates the UFS file systems as part of the installation process. Volume management products, like Solaris Volume Manager, create more sophisticated volumes, that expand beyond single slice or single disk boundaries hence we recommend using them. However during manual creation of the UFS file system using the ***newfs*** command, the following table shows the default parameters:

Parameter	Default Value & Tuning recommendation
Block size	8 Kbytes. For most UFS based CM systems, an 8-Kbyte file system block size provides the best performance, offering a good balance between disk performance and the use of space in primary memory and on disk. Ensure that if using Oracle® the **Database block size** is an integer multiple of this number when configuring the database, else performance will be impacted.
Fragment size	1 Kbyte. When choosing a fragment size, consider the trade-off between time and space: A small fragment size saves space, but requires more time to allocate. As a general rule, to increase storage efficiency, use a larger fragment size for file systems when most of the files are large. Use a smaller fragment size for file systems when most of the files are small. For multiterabyte file systems, the fragment size must be equal to the file system block size.
Minimum free space	((64 Mbytes/partition size) * 100), rounded down to the nearest integer and limited to between 1% and 10%, inclusively.
Number of inodes	1 inode for each 2 Kbytes of data space. Having too many inodes is much better than running out of inodes. If you have too few inodes, you could reach the maximum number of files on a disk slice that is practically empty.

Storage & I/O layout

Optimally configured I/O systems have a balanced choice of their physical hardware and logical configurations.

Tuning I/O can enhance the performance of the application if the I/O system is operating at or near capacity and is not able to service the I/O requests within an acceptable time. CM deployments that spend the majority of CPU time waiting for I/O activity to complete are usually I/O bound. Some storage device types are more optimal for CM applications

Selecting storage devices

Category	Options	Advantages
Hardware characteristics	1. Disk speed, size & type 2. The size of the cache 3. The number of channels and type of channels that are attached and online to a group of	Choosing disks with higher rotational speeds reduces random request service times. Solid-state drives while expensive

	logical volumes 4. The size of non-volatile storage (NVS), if deferred write performance is a problem 5. Fast Disk arrays 6. Advanced features such as Parallel Access Volumes (PAV), Multiple Allegiance, and FlashCopy®	typically consist of flash memory chips and have no moving parts such as spindle motors or rotating prats. The benefits that these drives offer are numerous, and include lower power consumption, enhanced performance, increased durability, less chance of data loss and no noise. For non SSD drives higher RPM means increased power consumption. Also, 2.5-inch drives consume less than half the power of older 3.5-inch drives.
RAID group size	Make sure your RAID groups are configured using the "magic" numbers 4+1 or 8+1.	These configurations are the fastest.
Mirror choices	Interaction of the write cache in the controller, the stripe size on the RAID groups, and the I/O size on your host.	

.

I/O profile

The I/O profile on the specific OS could be tuned to achieve greater performance. Some of the parameters that define Operating system's I/O profile are:

Parameter	Description
Transfer size	Number of bytes or blocks transferred.
Timing	The rate at which I/Os are generated including the timing difference between read and write operations.
Data Pattern	The data pattern filled into the write/read buffers and being sent to/from the disk device.
Processes	The amount of processes generating I/O operations that are running concurrently to a disk device. This includes the total number of hosts as well as the number of processes of I/O running to the end storage device.
Range	The area on which to execute the I/O and its size.
Seeking method	Sequential, random or mixed.
Throttling levels	SCSI layer's Queue Depth & SCSI Disk Timeout Value
Queue depth	Adjusted both in the OS block device parameters & on the Lower Layer of the SCSI Subsystem where the HBA throttles I/O with its own queue depth values.
Array	Stripe/chunk size of the disk device & its RAID type.
Stripe unit size	Optimal stripe unit size decreases with sequentiality and with good alignment between data boundaries and stripe unit boundaries. Optimal stripe unit size increases with concurrency, burstiness, and typical request sizes.
Storage controller cache and buffer resources	Configure to use the maximum available cache size. If the cache is substantially larger than the database buffer pools, database can make effective use of the cache to reduce I/O times for random I/O.

Choosing Data Block Size for Oracle®

A block size of 8K is optimal for most content manager implementations using Oracle®. However, OLTP systems occasionally use smaller block sizes. We tested with an 8 KB block size for Oracle® as it represents a good compromise and is usually effective.

Block Size Advantages and Disadvantages

Block Size	Advantages
Smaller	Good for small rows with lots of random access. Reduces block contention.
Larger	Lower overhead, hence more room to store data. Permits reading a number of rows into the buffer cache with a single I/O. Good for sequential access or very large rows.

Note, CM V8.4.2 now offers the option of choosing variable block sizes for Oracle® tables.

Lay Out the Files Using Operating System or Hardware Striping

If the OS has LVM software or hardware-based striping, then it is possible to distribute I/O using these tools. Options when using an LVM or hardware striping include **stripe depth** and **stripe width**.

- Stripe depth is the size of the stripe, sometimes called stripe unit.
- Stripe width is the product of the stripe depth and the number of drives in the striped set.

Reasonable stripe depths range from 256 KB to 1MB. We conducted our tests with a 512 KB stripe depth.

Deployment Type	Description	Advantages & usage
High-concurrency (random) systems	No single I/O request is broken up into more than one physical I/O call.	OLTP environments.
	Coarse grain striping.	Stripe depths are larger than the I/O size. Maximizes overall I/O throughput. multiblock reads, as in full table scans, will benefit when stripe depths are large and can be serviced from one drive.
	Oracle® databases.	The minimum stripe depth is twice the Oracle® block size.

I/O Layout options

If your system does not have an LVM or hardware striping, then I/O must be manually balanced across the available disks by distributing the files according to each file's I/O requirements. Below are some of the options for I/O Layout and configuration options for optimal performance.

Configuration type	Description	Advantage
Automatic Storage Management (ASM) (Oracle® 10g only)	Specifically for the storage of Oracle® database files	Can add more physical storage, change storage configurations, or remove extraneous storage without having to shut down Oracle® database to change out physical hardware. Provides LVM, fault tolerance & I/O load balancing out of the box.
Move Archive Logs to Different Disks	Capacity determined by the rate of archive log generation and the amount of archive storage required.	The archive can be performed at very high rate (using sequential I/O).
Place hot blocks on Separate Disks/ volumes	Move Database redo logs to specific fast disks	Redo logs are write-intensive. As each user *must* wait upon each commit until the before image information is written to the redo log, it is critical to make this process occur as fast as possible.
	Move Frequently accessed Oracle® or DB2 tables to dedicated disks or outermost tracks.	Storing "hot" data near the "beginning" of a disk yields significant performance benefits because this corresponds to the outermost (fastest) tracks.
	Move Library server & Resource manager CM log files to dedicated disks.	
	Move Resource Manager, LBOSDATA & staging directories to dedicated disks.	Writing of the LBOSDATA is not impaired by any other I/O.
Stripe Everything Across Every Disk	Build one large volume, striped across all available disks using RAID 1 or RAID 5.	Allows dynamic reconfiguration, which allows disks to be added while the system is online.

Database Tuning

IBM DB2®

This section provides information on tuning a DB2® Database Management System for optimal performance.

There are two sets of configuration parameters that will be discussed.

- The Database Manager Configuration, or DBM, parameters apply to the database instance.
- The Database Configuration, DB, parameters apply to the overall database configuration.

Tools for Monitoring & Tuning the DB2 Database

Tool	Usage/ Purpose
Snapshot monitor	Focuses on summary data, where tools like counters, timers, and histograms maintain running totals of activity in the system.
DB2 Performance Expert	Consolidates, reports, analyzes and recommends self-managing and resource tuning changes.
DB2 Health Center	Provides different methods to work with performance-related information and replace the performance monitor capability of the Control Center.

Important Initialization Parameters with Performance Impact:

DB2 initialization parameters fall into various functional groups. For example, parameters perform the following functions:

- Set limits for the entire database
- Set user or process limits
- Set limits on database resources that affect performance (these are called variable parameters). Variable parameters are of particular interest to database administrators, because these parameters are used primarily to improve database performance.

Under some scenarios, DB2v10.x might exhibit performance degradation (see DB2 APARs IC94359 and IC94360 for details). Before fix available, you can keep the registry variable turn ON as workaround.

db2set -im DB2COMPOPT=327685,524288

If DB2TS feature enabled, you may need set DB2 registry variable (MALLOCOPTIONS) for better performance. Refer to the link for more detail information.

http://www-01.ibm.com/support/docview.wss?uid=swg21610038

Autonomic Performance Tuning Settings

DB2 V9.7 & V10.x with the use of the built in autonomics for database tuning is recommended for use with Content Manager V8.5.

To learn more about DB2 Autonomic features please reference the DB2 V9.7, V10.1 or V10.5 Infocenter and search for the term “Autonomic computing overview”

Refer to DB2 solution information center at:
http://publib.boulder.ibm.com/infocenter/db2luw/v9r7//index.jsp
http://publib.boulder.ibm.com/infocenter/db2luw/v10r1//index.jsp
http://publib.boulder.ibm.com/infocenter/db2luw/v10r5//index.jsp

Each Content Manager implementation is different and workloads vary. While some initial settings are shown below, it is recommended that customers enable the autonomic features in DB2 and monitor the system to determine if adjustments are appropriate. The db2diag.log and the stmm log contain entries when adjustments are made. These logs can be parsed to determine minimum and maximum sizes set for any parameter. If it is determined a parameter has a very small range then it may be appropriate to manually set that parameter to the high end of the range and eliminate the constant adjustments. More information on stmm can be found in the DB2 Infocenter by searching for “Self tuning memory operational details and limitations”

A self-tuning memory manager log parser available at
http://www.ibm.com/developerworks/data/library/techarticle/dm-0708naqvi/index.html

Library Server DB2 instance:

When the instance is started it will determine available memory and set these parameters. No change is recommended for these parameters at this time.

Parameter	Setting
Database monitor heap size (4KB)	(MON_HEAP_SZ) = AUTOMATIC
Max number of coordinating agents	(MAX_COORDAGENTS)= AUTOMATIC
Max number of client connections	(MAX_CONNECTIONS) = AUTOMATIC
Size of instance shared memory (4KB)	(INSTANCE_MEMORY) = AUTOMATIC
No. of int. communication buffers(4KB)	(FCM_NUM_BUFFERS) = AUTOMATIC
No. of int. communication channels	(FCM_NUM_CHANNELS) = AUTOMATIC

The number of pool agents should be set to an initial number for each customer's unique workload. The customer should monitor the number of agents being created and destroyed and set this parameter to an appropriate value for an average workload.

Parameter	Setting
Agent pool size	(NUM_POOLAGENTS) = AUTOMATIC(100)

Library Server Database

By setting these parameters to Automatic and using the system default configuration you allow DB2 to balance the allocation of memory based on the workload. No change is recommended at this time.

Parameter	Setting
Size of database shared memory (4KB)	(DATABASE_MEMORY) = AUTOMATIC
Package cache size (4KB)	(PCKCACHESZ) = AUTOMATIC
Application Memory Size (4KB)	(APPL_MEMORY) = AUTOMATIC
Statistics heap size (4KB)	(STAT_HEAP_SZ) = AUTOMATIC
Number of asynchronous page cleaners	(NUM_IOCLEANERS) = AUTOMATIC
Number of I/O servers	(NUM_IOSERVERS) = AUTOMATIC
Default prefetch size (pages)	(DFT_PREFETCH_SZ) = AUTOMATIC

Content Manager is a unique application in that is completely written using stored procedures. For this environment it has been determined that setting the DBHEAP and APPLHEAPSZ to these initial settings is appropriate. Each workload is different. The DBHEAP size should be monitored and tuned based on recommendations made by DB2.

Parameter	Setting
Database heap (4KB)	(DBHEAP) = AUTOMATIC(2400)
Default application heap (4KB)	(APPLHEAPSZ) = AUTOMATIC(512)

The default heap size for sort (SORTHEAP) is 256 pages, a size that is equivalent to 1 MB (with each page as 4 KB). Depending on your application, you can add more pages to it, but do not add more than the sort threshold size for the instance (SHEAPTHRES).

The LOCKLIST and MAXLOCKS parameters should not be set to Automatic in your Content Manager environment. The LOCKLIST parameter should be sized for each customer workload. Every workload is different and LOCKLIST usage should be monitored over time to determine the correct setting.

Parameter	Setting
Max storage for lock list (4KB)	(LOCKLIST) = ####
Percent. Of lock lists per application	(MAXLOCKS) = 50

This table shows the initial Automatic Maintenance settings. It is recommended that the Automatic maintenance, Automatic table maintenance and Automatic runstats parameters be set to ON. The customer may decide to enable other Automatic Maintenance functions based on their needs.

Parameter	Setting
Automatic maintenance	(AUTO_MAINT) = ON
Automatic database backup	(AUTO_DB_BACKUP) = OFF
Automatic table maintenance	(AUTO_TBL_MAINT) = ON
Automatic runstats	(AUTO_RUNSTATS) = ON
Automatic statement statistics	(AUTO_STMT_STATS) = OFF
Automatic statistics profiling	(AUTO_RUNSTATS) = ON
Automatic profile updates	(AUTO_PROF_UPD) = OFF
Automatic reorganization	(AUTO_REORG) = OFF
Enable XML Character operations	(ENABLE_XMLCHAR) = YES

Resource Manager Instance

For the Resource Manager, the default settings assigned during instance creation for the following parameters are recommended.

Parameter	Setting
Database monitor heap size (4KB)	(MON_HEAP_SZ) = AUTOMATIC
Size of instance shared memory (4KB)	(INSTANCE_MEMORY) = AUTOMATIC
Agent pool size	(NUM_POOLAGENTS) = AUTOMATIC
Max number of coordinating agents	(MAX_COORDAGENTS) = AUTOMATIC
Max number of client connections	(MAX_CONNECTIONS) = AUTOMATIC
Number of pooled fenced processes	(FENCED_POOL) = AUTOMATIC
No. of int. communication buffers(4KB)	(FCM_NUM_BUFFERS) = AUTOMATIC
No. of int. communication channels	(FCM_NUM_CHANNELS) = AUTOMATIC

Resource Manager Database

Several parameters are set to Automatic during the database creation. No change is recommended at this time.

Parameter	Setting
Size of database shared memory (4KB)	(DATABASE_MEMORY) = AUTOMATIC
Package cache size (4KB)	(PCKCACHESZ) = AUTOMATIC
Sort list heap (4KB)	(SORTHEAP) = AUTOMATIC
SQL statement heap (4KB)	(STMTHEAP) = AUTOMATIC
Application Memory Size (4KB)	(APPL_MEMORY) = AUTOMATIC
Statistics heap size (4KB)	(STAT_HEAP_SZ) = AUTOMATIC
Number of asynchronous page cleaners	(NUM_IOCLEANERS) = AUTOMATIC
Number of I/O servers	(NUM_IOSERVERS) = AUTOMATIC
Default prefetch size (pages)	(DFT_PREFETCH_SZ) = AUTOMATIC
Average number of active applications	(AVG_APPLS) = AUTOMATIC

For Content Manager the default DBHEAP and APPLHEAPSZ are appropriate.

Parameter	Setting
Database heap (4KB)	(DBHEAP) = AUTOMATIC(1000)
Default application heap (4KB)	(APPLHEAPSZ) = AUTOMATIC(512)

This table shows the initial Automatic Maintenance settings. As for the Library Server database, we recommend that the Automatic maintenance, Automatic table maintenance and Automatic runstats parameters be set to ON. The customer may decide to enable other Automatic Maintenance functions based on their needs.

Parameter	Setting
Automatic maintenance	(AUTO_MAINT) = ON
Automatic database backup	(AUTO_DB_BACKUP) = OFF
Automatic table maintenance	(AUTO_TBL_MAINT) = ON
Automatic runstats	(AUTO_RUNSTATS) = ON

Automatic statement statistics	(AUTO_STMT_STATS) = OFF
Automatic statistics profiling	(AUTO_STATS_PROF) = OFF
Automatic profile updates	(AUTO_PROF_UPD) = OFF
Automatic reorganization	(AUTO_REORG) = OFF
Enable XML Character operations	(ENABLE_XMLCHAR) = YES

If your Library Server is using 64-bit stored procedures consider using "db2set DB2_AGENT_CACHING_FMP=ON" to retain an active db2fmp processes for each connection. If the cache begins to exceed your system memory, then db2set DB2_AGENT_CACHING_FMP=OFF to resume pooling db2fmp processes.

Memory allocation & Buffer pool management

In all environments, the amount of physical memory limits how many DB2 agent processes can be created. Each agent process occupies a certain number of megabytes of physical memory, so the total amount of memory used by all DB2 processes can be calculated roughly as the sum of memory used by the db2syscs.exe and db2fmp processes.

Operating System	db2fmp process memory footprint
AIX	5 MB
Windows 2012 and 2008	15 MB

A bufferpool is the memory area that is used to cache table and index data pages as they are being read from disk or being modified. The buffer pool improves database system performance by enabling data to be accessed from memory instead of from disk. Because most data manipulation takes place in buffer pools, the configuration of buffer pools is the single most important tuning area. Only large objects and long field data are not manipulated in a buffer pool.

Tuning tips for Buffer Pool

Increasing the buffer pool size generally improves the hit ratio, but you will reach a point of diminishing returns. Ideally, if you can allocate a buffer pool large enough to store your entire database you would get a hit ratio of 100% when the system is up and running. However, this is unrealistic in most cases. The significance of the hit ratio depends on the size of your data and the way it is accessed.

For a large database, increasing the buffer pool size might have minimal effect on the buffer pool hit ratio. The number of data pages can be so large that the statistical chances of a hit are not improved by increasing its size. In this case, tuning the index pool hit ratio could achieve the desired result. This can be done using two methods:

- Split the data and indexes into two different buffer pools and tune them separately.
- Use one buffer pool, but increase its size until the index hit ratio stops increasing.

A very large database where data is accessed evenly would have a poor hit ratio. There is little you can do with very large tables. In such a case, you should focus on smaller, frequently accessed tables, and on the indexes- perhaps assigning them to individual buffer pools, in which you can aim for higher hit ratios.

In CM, we recommend that buffer pools be sized to keep all system tables that contain definitions (i.e. ItemTypes, ComponentTypes, attributes) in memory to avoid disk I/O. You should adjust your buffer pool size according to the resources available and workload.

DB2 Connection Concentrator

For DB2 with the use of the Connection Concentrator is NOT recommended.
With previous versions of DB2 it was sometimes recommend for use. The Connection Concentrator is off when the MAX_COORDAGENTS and MAX_CONNECTIONS are set equal.

Please see the Content Manager technote at:
http://www-01.ibm.com/support/docview.wss?uid=swg21319866 for recommendations on when to use the DB2 Connection Concentrator

Configuration parameters that affect query optimization

Runstats/rebind

When the SQL compiler optimizes SQL query plans, its decisions are heavily influenced by statistical information about the size of database tables and indexes. The optimizer also uses information about the distribution of data in specific columns of tables and indexes if these columns are used to select rows or join tables. It uses this information to estimate the costs of alternative access plans for each query. So it is very important to keep statistics current. When significant number of table rows are added or deleted, or if data in columns for which you collect statistics is updated, execute the *runstats* command to update the statistics.

Statistical information is collected for specific tables and indexes in the local database when you execute the *runstats* utility. The collected statistics are stored in the system catalog tables.

Dynamic SQL statements will automatically use these updated statistics, but static SQL statements built inside CM require the *db2rbind* command to run before they can use the updated statistics. The db2rbind command revalidates all packages in the database.

Execution of ***runstats***/db2***rbind*** should be done as part of your regular database maintenance. As your Library Server and Resource Manager databases grow and change, recalculating table statistics is critical to improving database performance.

The following is a shell script that dynamically determines the Library Server database tables that are candidates for *runstats*, performs *runstats* on those tables, then does a *db2rbind* to complete the maintenance process:

```
db2 connect to icmnlsdb user <uid> using <password>

db2 -x "select tabname from syscat.tables where type='T' and
tabschema='ICMADMIN'" | sed -e "s/ *.$//" | sed -e "s/^/runstats
on table ICMADMIN./" | sed -e "s/$/ with distribution and detailed
indexes all;/" > dostatsls.sql

db2 -tvf dostatsls.sql
db2rbind icmnlsdb -l icmnlsdb.log all -u <uid> -p <password>
```

The *dostatsls.sql* file should contain lines similar to the following:

```
runstats on table ICMADMIN.ICMSTADMINDOMAINS with distribution and detailed indexes all;
runstats on table ICMADMIN.FAUSERMAP with distribution and detailed indexes all;
runstats on table ICMADMIN.ICMSTPRIVDEFS with distribution and detailed indexes all;
runstats on table ICMADMIN.FALOG with distribution and detailed indexes all;
```

Similarly, the following is a shell script that dynamically determines the Resource Manager database tables that are candidates for *runstats*, runs *runstats* against those tables, then does a *db2rbind*:

```
db2 connect to rmdb user <uid> using <password>

db2 -x "select tabname from syscat.tables where type='T' and
tabschema='RMADMIN'" | sed -e "s/ *.$//" | sed -e "s/^/runstats on
table RMADMIN./" | sed -e "s/$/with distribution and detailed
indexes all;/" > dostatsrm.sql

db2 -tvf dostatsrm.sql
db2rbind rmdb -l rmdb.log all -u <uid> -p <password>
```

The *dostatsrm.sql* file should look similar to the above *dostatsls.sql* file.

Tables/Indexes reorganization

After many changes to table data, logically sequential data may be on non-sequential physical data pages so that the database manager must perform additional read operations to access data. Additional read operations are also required if a significant number of rows have been deleted. In such a case, you might consider reorganizing the table to match the index and to reclaim space. You can reorganize the system catalog tables as well as database tables.

Since reorganizing tables and indexes is costly, you should run the ***reorgchk*** command to make sure if there is a need to do the tables and indexes reorganization. An example of how to use the command is as following:

```
db2 connect to <database> user <uid> using <password>
db2 reorgchk current statistics on table all
```

```
Table statistics:

F1: 100 * OVERFLOW / CARD < 5
F2: 100 * (Effective Space Utilization of Data Pages) > 70
F3: 100 * (Required Pages / Total Pages) > 80

SCHEMA    NAME                  CARD    OV    NP    FP ACTBLK    TSIZE  F1  F2  F3 REORG
-------------------------------------------------------------------------------------
RMADMIN   RMACCESS                 2     0     1     1      -      386   0   - 100 ---
RMADMIN   RMBLOBS                  -     -     -     -      -        -   -   -   - ---
RMADMIN   RMCNTL                   1     0     1     1      -      225   0   - 100 ---
RMADMIN   RMCOLLECTIONS           16     0     1     1      -      992   0   - 100 ---
RMADMIN   RMDEVMGR                 6     0     1     1      -      540   0   - 100 ---
RMADMIN   RMMGTCLASS               1     0     1     1      -       44   0   - 100 ---
RMADMIN   RMMGTTRANSITION          1     0     1     1      -       20   0   - 100 ---
RMADMIN   RMOBJECTS           611242     0  3748  3748      - 1.22e+08   0  99 100 ---
RMADMIN   RMPARTS                  -     -     -     -      -        -   -   -   - ---
RMADMIN   RMREPLICATION            -     -     -     -      -        -   -   -   - ---
RMADMIN   RMSERVER                 2     0     1     1      -      750   0   - 100 ---
RMADMIN   RMSTAGING                1     0     1     1      -       57   0   - 100 ---
RMADMIN   RMSTGGRPCLASS           16     0     1     1      -      224   0   - 100 ---
RMADMIN   RMSTGGRPVOLUME          16     0     1     1      -      256   0   - 100 ---
RMADMIN   RMSTORAGECLASS           1     0     1     1      -       57   0   - 100 ---
```

You should pay attention to the last column (REORG). Each hyphen (-) displayed in this column indicates that the calculated results were within the set bounds of the corresponding formula. Each asterisk (*) indicates that the calculated results exceeded the set bounds of its corresponding formula.

- - or * on the left side of the column corresponds to F1 (Formula 1)
- - or * in the middle of the column corresponds to F2 (Formula 2)
- - or * on the right side of the column corresponds to F3 (Formula 3)

Table reorganization is suggested when the results of the calculations exceed the bounds set by the formula. For example, --- indicates that, since the formula results of F1, F2, and F3 are within the set bounds of the formula, no table organization is suggested. The notation *-* indicates that the results of F1 and F3 suggest table reorganization, even though F2 is still within its set bounds.

For more information, please refer to DB2 Information Center.

Create Additional Indexes

When a new Item Type is defined you should consider how the attributes will be searched for and create the appropriate indexes. Over time the customer use patterns may change and different attributes queried against resulting in slow response times. The event monitor and snapshots help in determining which queries are performing badly and may be candidates for new indexes.

Tuning DB2 transaction log characteristics:

The log buffer is an area of memory that helps speed up the database logging process. DB2 writes information about all transactions to the log buffer and then flushes this buffer to disk

periodically. This improves database performance since DB2 does not have to write every change to disk immediately when it makes changing in the database.

DB2 writes the log buffers to disk when either of these occurs:

- One or more transactions commit
- The log buffer is full
- One second has elapsed since the last log buffer flush

To improve performance, you can tell DB2 to wait to flush the log buffer until a specific number of commits occur. This number is called the *number of commits to group* and is controlled by the *mincommit* database configuration parameter. When you set this parameter to a value greater than one, the applications that issue a commit may not return immediately because DB2 must ensure that the log buffer is written to disk before returning to the application.

Since the log buffer is also flushed to a disk when it becomes full, it is important to have a log buffer that is large enough. This would eliminate the log buffer from constantly writing to the disk.

You should put the database log files on a different disk that does not have high I/O. This allows efficient logging activities with a minimum overhead of waiting for I/O.

DB2 related settings/ exceptions for Windows 2008

On Windows 2008 platform, we encountered SQLCODE=-1042 error message while configuring Library Server. The solution is as below:

- Uncheck "Enable operation system security" on the page “Enable Operating System security for DB2 objects” during DB2 installation.
- Turn off db2 extended security "db2set DB2_EXTSECURITY=NO"

Initial Tablespace Definitions

The Content Manager cnfiguration utility creates a default set of tablespaces and bufferpools when creating the Library Server and Resource Manager databases.

- For high volume Item Types you should consider creating additional tablespaces to contain them and achieve higher performance.

  Search the CM Information Center at http://publib.boulder.ibm.com/infocenter/cmgmt/v8r5m0/index.jsp for the phrase “deferred DDL” for guidance on how to do this.

Library Server Database – DB2 tablespaces

Bufferpool	Tablespace	Description
ICMLSMAINBP32	ICMLFQ32	Holds all the large, persistent, "frequently used" tables - most notably, all the item type tables. When a new item type is defined, the system administration client defines the new tables in this tablespace. If you customize the DDL, this tablespace should be defined across multiple containers for performance and scalability.

		-Default for all the Item Type tables (32 KB page size)
ICMLSMAINBP32	ICMLNF32	Holds the large but less-frequently used tables, such as event logs, versions, and replicas. If you don't use any of these features, this tablespace won't need much space. - Event logs, versions and replicas. (32 KB page size)
ICMLSVOLATILEBP4	ICMVFQ04	Holds the "volatile" tables, those whose size is usually small but which can vary widely, and for which both updates and deletes are very common. For example, the tables associated with document routing "in progress" items, checked out items, etc. Putting this on a separate physical disk should help performance. -- Document routing "in process" items, Checked out item, etc. (4 KB page size)
ICMLSFREQBP4	ICMSFQ04	Holds all the small "frequently used" but seldom updated tables, such as system information, user definitions, ACLs, federated admin, etc, that don't take up much space but that are always needed in memory for good performance. – ACLs, user definitions, system information, etc (4 KB page size)
CMBMAIN4	CMBINV04	Holds the federated inventory tables. If you don't use federated access, this tablespace will be very small. Holds the tables used for II4C federated administration (4 KB page size) (deprecated)
ICMLSVOLATILEBP4	ICMLSSYSTSPACE4	Temporary tables (4 KB page size)
ICMLSMAINBP32	ICMLSYSTSPACE32	Temporary tables (32KB page size)
ICMLSMAINBP32	ICMLOB32	BPM event monitor tables(32 KB page size)
IBMDEFAULTBP	ICMLOBCOL4	BPM event monitor tables (4 KB page size)

Resource Manager Database – DB2 tablespaces

Bufferpool	Tablespace	Description
OBJECTPOOL	OBJECTS	Holds table that grows very large. – Information about objects stored on this Resource Manager (32 KB page size)
OBJPARTSPOOL	OBJPARTS	(32 KB page size)
SMSPOOL	SMS	Holds small but frequently read administrative information (4 KB page size)
TRACKINGPOOL	TRACKING	Holds volatile tables related to transactions (4 KB page size)
VALIDATEPOOL	VALIDATEITM	Holds tables used by he validation utility – infrequently used but can create a very large volume of data (32 KB page size)
REPLICAPOOL	REPLICAS	Holds tables for replication (32 KB page size)
BLOBPOOL	BLOBS	Objects stored in Database (32 KB page size)

Oracle® 11gR2

This section provides information on tuning an Oracle® 11gR2 Database system for optimal CM performance.

The key tools to measure performance and subsequently tune system & database parameters include the following tools:

Tools for Monitoring & Tuning the Oracle® 11gR2 Database

Tool	Usage/ Description
Oracle® Enterprise Manager	Provides an overall web based dashboard for Monitoring & performance advisors and for Oracle® utilities.
Oracle® Diagnostics Pack	Provides unique features, such as automatic identification of performance bottlenecks, guided problem resolution, and comprehensive system monitoring via tools such as Automatic Database Diagnostics Monitor (ADDM) and the Automatic Workload Repository (AWR).
Oracle® Tuning Pack	Automates the entire database application tuning process. Key tools include the SQL Tuning Advisor and the SQL Access Advisor.

Detecting Common Performance Problems in Oracle® Databases

While there could be an entire gamut of issues related with database performance; this section lists and describes common performance problems found in Oracle® databases & choices available to fix issues.

Problem	Tool used	Tuning
CPU bottlenecks	ADDM & OEM – Performance summary page	On the Performance Summary page, verify the top processes in the Top 10 Processes section. Next, use Oracle® Database Resource Manager to reduce the impact of peak-load-use patterns by prioritizing CPU resource allocation.
Memory bottlenecks	OEM – **Performance** page & Automatic Shared Memory Management to manage the SGA memory	Reduce the number of open cursors and hard parsing with cursor sharing.
	Automatic PGA Management to manage SQL memory execution	Reduce the number of processes that use a lot of memory.
Disk I/O utilization	**Performance Summary** page, from the **View** list, select **Disk Details**	Stripe across every disk to distribute I/O.
		Identify hot spots on files & Moving files, such as archive logs and redo logs, to separate disks
		Store required data in memory to reduce the number of physical I/Os.

Memory allocation & Buffer pool management

In Oracle environments, the amount of physical memory limits how many Oracle external procedures can be created. Each agent process occupies a certain amount of physical memory, so the total memory used by all external procedures can be roughly calculated as the sum of memory used by the external procedures.

Oracle 11gR2 Automatic Shared Memory Management

The automatic shared memory management feature of Oracle11gR2 simplifies administration of the various memory components that make up the System Global Area (SGA). Oracle11gR2 administrators are able to set a single dynamic parameter, the SGA_TARGET, to allocate memory to Oracle SGA. Oracle11gR2 uses this value contained in SGA_TARGET to determine the memory to be allocated to the buffer cache, shared pool, large pool and java pool. If the SGA_TARGET is set to a non-zero value and the STATISTICS_LEVEL initialization parameter is set to TYPICAL or ALL, the database will perform automatic memory reallocation. Based on tests conducted in the IBM CM Performance Labs, we strongly suggest using automatic shared memory management feature of Oracle11gR2.

Performance Considerations for Initial Instance Configuration:

IBM CM 8.5 requires you to use the Oracle Database Configuration Assistant (DBCA) tool to create a database instance. The supplied seed database templates include the necessary basic initialization parameters. A few parameters may need tweaking to meet the performance requirements at your site. Initialization parameter values apply to the entire database, not to an individual user, unless otherwise specified.

Initialization parameters fall into various functional groups. For example, parameters perform the following functions:

- Set limits for the entire database
- Set user or process limits
- Set limits on database resources that affect performance (these are called variable parameters). Variable parameters are of particular interest to database administrators, because these parameters are used primarily to improve database performance.

Important Oracle Initialization Parameters with Performance Impact:

Init parameter	Description
sga_max_size	Initial size of SGA at startup
db_block_size	Specifies the size of the Oracle database blocks stored in the database files and cached in the SGA.
sga_target	Specifies the total size of all SGA components. We recommend using Automatic Shared Memory Management, to enable Oracle® dynamically reconfigure the sizes of the shared pool, the large pool, the buffer cache,

	and the process-private memory.
Processes	Maximum number of operating system user processes that can simultaneously connect to Oracle. Its value should allow for all background processes such as locks, job queue processes, and parallel execution processes.
db_writer_processes	1 or CPU_COUNT / 8, whichever is greater
statistics_level	When the STATISTICS_LEVEL parameter is set to ALL, additional statistics are added to the set of statistics collected with the TYPICAL setting.
open_cursors	Set the value of OPEN_CURSORS high enough to prevent your application from running out of open cursors within the memory availability.
optimizer_mode	Choose first_rows_100; The optimizer uses a cost-based approach and optimizes with a goal of best response time to return the first 100 rows.
undo_management	AUTO mode is recommended.
filesystemio_options	Enable or disable asynchronous I/O or direct I/O on file system files. This parameter is platform OS specific.
	COMMIT_WRITE is an advanced parameter used to control how redo for transaction commits is written to the redo logs. The IMMEDIATE and BATCH options control how redo is batched by Log Writer. The WAIT and NOWAIT options control when the redo for a commit is flushed to the redo logs.

Tablespaces:

Table space	Tuning & Description
Undo	We recommend running in automatic undo management mode as it transparently creates and manages undo segments.
Temporary	We recommend the use of locally managed temporary tablespaces with a UNIFORM extent size of 1 MB as it helps optimize disk sort performance.
Application tables and indexes	Set automatic segment-space management for tablespaces.

Sizing Redo Log Files:

The size & placement location of the redo log files influences performance, because the behavior of the database writer and archiver processes depend on the redo log sizes. Generally, larger redo log files provide better performance.

The optimal size can be obtained by querying the OPTIMAL_LOGFILE_SIZE column from the V$INSTANCE_RECOVERY view. You can also obtain sizing advice on the Redo Log Groups page of Oracle® Enterprise Manager Database Control. A rough guide is to switch logs at most once every twenty minutes.

Log file sync waits occur when sessions wait for redo data to be written to disk. Typically this is caused by slow writes or committing too frequently in the application. Some recommendations are:

- Tune LGWR to get good throughput to disk . Place the redo logs on your fastest disks, different disks separate from other datafiles.
- On the Resource Manager database : ALTER SYSTEM SET COMMIT_WRITE = 'IMMEDIATE,NOWAIT';

COMMIT_WRITE is an advanced parameter used to control how redo for transaction commits is written to the redo logs. The IMMEDIATE option controls how redo is batched by Log Writer. The NOWAIT option controls when the redo for a commit is flushed to the redo logs.

Planning the Network :

Oracle*Net is a layer in the OSI model that resides above the network-specific protocol stack Oracle Net Services provide a variety of options to help you design and manage networks that are both flexible and easy to use. Oracle Net Services offers a number of features that can help reduce round-trip time across the network, increase listener performance, and reduce the number of protocols used.

Parameter	Description
Listener Queue Size	This parameter is used for high-volume databases, where the listener spawns thousands of connections per hour. The number of expected simultaneous connections should be equal to the size of the *queuesize* parameter. A disadvantage of this parameter is that it pre-allocates resources for anticipated requests, therefore using more system memory and resources.
Session Data Unit Size for Data Transfer Optimization	Consider changing the SDU size when the predominant message size is smaller or larger than 2048. The SDU size you choose should be 70 bytes larger than the predominant message size, as long as the maximum SDU size is not exceeded. If the predominant message size plus 70 bytes exceeds the maximum SDU, then the SDU should be set such that the message size is divided into the smallest number of equal parts where each part is 70 bytes less than the SDU size. The SDU size can range from 512 bytes to 32767 bytes.
tcp.nodelay parameter in protocol.ora	Oracle*Net, by default, waits until the buffer is full before transmitting data. Therefore, requests are not always sent immediately to their destinations.
	The *protocol.ora* file can be specified to indicate no data buffering for all TCP/IP implementations.

Oracle® Sql*Net & TCP Tuning :

Even with a fast physical network, it is possible for clients and servers to run inefficiently. This can happen when the logical and physical sizes of packets do not match and packets are split at the NIC or SQL*Net layer.

Oracle® sets the session data unit (SDU) in Sql*net to 2048 bytes by default. This works fine if the data being delivered is less than 2048 bytes. If the amount of data to be delivered is greater than the SDU, the Oracle® client and server splits it into multiple packets. The Oracle® server process consequently will incur a wait event that causes it to context switch and wait until more data is available. Sql*net wait events are found in StatsPack reports.

Parameter	Description & Tuning
Maximum transport unit (MTU)	Value supported by the OS or network hardware.

Transport data unit (TDU)	Oracle® Sql*Net layer will store data up to this value before transferring to the SDU.
Session data unit (SDU)	Packets sent from Sql*Net to the OS will be no bigger than this size. It is good practice to set the Oracle® SDU to a multiple of the supported MTU. Oracle® recommends setting this parameter in both the client-side and server-side sqlnet.ora file to ensure the same SDU size is used throughout a connection. When the configured values of client and database server do not match for a session, the lower of the two values is used. DEFAULT_SDU_SIZE=8KB
TCP_TIME_WAIT_INTERVAL	Resource manager: Notifies TCP/IP on how long to keep the connection control blocks closed. After the applications complete the TCP/IP connection, the control blocks are kept for the specified time. When high connection rates occur, a large backlog of the TCP/IP connections accumulates and can slow server performance. The default time wait interval for a Solaris operating system is 240000 milliseconds. Recommended value: 30000 milliseconds
TCP_FIN_WAIT_2_FLUSH_INTERVAL	Resource manager: Specifies the timer interval prohibiting a connection in the FIN_WAIT_2 state to remain in that state. When high connection rates occur, a large backlog of TCP/IP connections accumulates and can slow server performance. Default value: 675000 milliseconds Recommended value: 67500 milliseconds
Connection backlog	Resource manager: when a high rate of incoming connection requests result in connection failures: Recommended tcp_conn_req_max_q value: 8000

Automatic Statistics Gathering:

Statistics are important for the query optimizer. The recommended approach to gathering statistics is to allow Oracle® to automatically gather the statistics. Oracle® 11gR2 gathers statistics on all database objects automatically and maintains those statistics in a regularly-scheduled maintenance job. Optimizer statistics are automatically gathered with the job GATHER_STATS_JOB.

This monitoring is enabled by default when STATISTICS_LEVEL is set to TYPICAL or ALL which is preset in the default templates shipped with CM 8.5. The GATHER_DATABASE_STATS or GATHER_SCHEMA_STATS procedures gather new statistics for tables with stale statistics when the OPTIONS parameter is set to GATHER STALE or GATHER AUTO. If a monitored table has been modified more than 10%, then these statistics are considered stale and gathered again.

We recommend using the Automatic Workload Repository and Automatic Database Diagnostic Monitor for statistics gathering, monitoring, and tuning due to the extended feature list and accuracy of results obtained during our benchmark testing. However If your site does not have the Automatic Workload Repository and Automatic Database Diagnostic Monitor features enabled, then use Statspack at levels greater than 7 to gather Oracle® instance statistics.

Automatic Performance Tuning Features:

The Oracle® automatic performance tuning features include:

Feature	Description
OEM advisors	The memory advisors are commonly used when automatic memory management is not set up for the database. Other advisors are used to optimize mean time to recovery (MTTR), shrinking of segments, and undo tablespace settings.
Automatic Database Diagnostic Monitor (ADDM)	Analyzes the information collected by the AWR for possible performance problems with the Oracle® database.
SQL Tuning Advisor	A quick and efficient technique for optimizing SQL statements without modifying any statements.
Automatic Workload Repository (AWR)	Collects, processes, and maintains performance statistics for problem detection and self-tuning purposes.
Server-generated alerts	Automatically provides notifications when impending problems are detected.

The Automatic Database Diagnostic Monitor (ADDM) is a self-diagnostic engine built into Oracle® Database. ADDM examines and analyzes data captured in the Automatic Workload Repository (AWR) to determine possible performance problems in Oracle® Database. An ADDM analysis is performed after each AWR snapshot (every hour by default), and the results are saved in the database, which can then be viewed using Oracle® Enterprise Manager.

Based on our test results, IBM's CM Performance team recommends using Oracle's Automatic Database Diagnostic Monitor (ADDM) on an ongoing basis & following ADDM advisor prior to making any changes to the database. The ADDM analysis is performed from the top down, first identifying symptoms and then refining the analysis to reach the root causes of performance problems.

Tuning WebSphere Application Server V8

WebSphere Application Server provides tunable settings for its major components to enable you to make adjustments to better match the runtime environment to the characteristics of your Resource Manager. The Resource Manager might need changes, for example, a larger heap size, to achieve optimal performance.

Tuning WebSphere Application Server is a critical part of getting the best performance from your Resource Manager. However, tuning WebSphere Application Server involves analyzing performance data and determining the optimal server configuration. This determination requires considerable knowledge about the various components in the application server and their performance characteristics.

The following steps are involved during the iterative tuning process:

a. Process definition & Identify constraints
b. Specify the load & Tune the application serving environment
c. Set performance goals & Analyze the performance data with performance advisors

Obtaining advice from the advisors

Advisors provide a variety of recommendations that help improve the performance of your application server & identify constraints.

Performance advisor types and purposes

The performance advisors encapsulate Application server knowledge, analyze the performance data, and provide configuration recommendations to improve the application server performance. Therefore, the performance advisors provide a starting point to the application server tuning process and help you without requiring that you become an expert.

Two performance advisors are available: the Runtime Performance Advisor and the performance advisor in Tivoli Performance Viewer.

The Runtime Performance Advisor runs in the Java virtual machine (JVM) process of application server; therefore, it does not provide extensive advice. In a stand-alone application server environment, the performance advisor in Tivoli Performance Viewer runs within the application server JVM.

The performance advisor in Tivoli Performance Viewer (TPV) provides advice to help tune systems for optimal performance and provide recommendations on inefficient settings by using collected Performance Monitoring Infrastructure (PMI) data.

The following chart shows the differences between the Runtime Performance Advisor and the Tivoli Performance Viewer advisor:

	Runtime Performance Advisor	Tivoli Performance Viewer
Start location	Application server	Tivoli Performance Viewer client
Invocation of tool	Administrative console	Tivoli Performance Viewer
Output	• The SystemOut.log file • The administrative console	Tivoli Performance Viewer in the administrative console
Frequency of operation	Configurable	Select refresh in the Tivoli Performance Viewer administrative console
Types of advice	Performance advice: • Web container thread pools • Connection pool size • Persisted session size and time • Prepared statement cache size • Session cache size • Memory leak detection	Performance advice: • Web container thread pools • Connection pool size • Persisted session size and time • Prepared statement cache size • Session cache size • Dynamic cache size • Java virtual machine (JVM) heap size • DB2 Performance Configuration wizard

Tuning the application serving environment

Use the following options to tune the component behavior:

(JVM) heap sizes	In the majority of cases you should set the maximum JVM heap size to value higher than the initial JVM heap size. When the thresholds are reached, the garbage collector gets invoked to free up unused storage. Select **Servers -> Application Servers -> server1 -> Java and Process Management -> Process Definition -> Java Virtual Machine -> Maximum Heap Size** • Default Maximum Heap Size = (256) • Recommended Maximum Heap Size = (1024)
JDBC data sources and associated connection pools	Connection pooling can improve the response time of the CM application which requires multiple connections simultaneously. Select **Resources -> JDBC ->Data Sources -> icmrm_database -> Connection**

	Pool Properties • Connection Timeout = 180 Maximum = 500 (Default = 150) Minimum = 5 (Default = 5)
Web container	Thread pool size: This parameter controls the number of threads in a Web container. A high number of threads allows the Web container to serve more concurrent requests from the Web server and hence improves the performance of the system. But it also increases CPU usage and hence limits the scalability of the system Select **Servers -> Application Servers -> server1 -> Thread Pools -> WebContainer** • Minimum = 10 Maximum = 500 (Default = 50) Timeout = 3500
Timer managers	A timer manager acts as a thread pool for application components that use asynchronous beans. Select **Resources > Asynchronous beans > Timer managers**. • Maximum number of threads = 20 (default 10)
Work Managers	A work manager is a thread pool created for J2EE applications that use asynchronous beans. **Select Resources -> Asynchronous Bean -> Work Managers -> icmrm_RMWorkManager** • Maximum number of threads = 40 (default 20)

Appendix A: References

1. Performance Tuning for Content Manager, Redbooks
2. CM v8.4 Performance Monitoring & Troubleshooting Guide
   http://www-01.ibm.com/support/docview.wss?uid=swg27017876
3. CM v8.4 Performance Tuning Guide
   http://www-01.ibm.com/support/docview.wss?uid=swg27016780
4. Understanding DB2: Learning Visually with Examples, Second Edition
5. Oracle® Database (RDBMS) on Unix AIX,HP-UX,Linux,Mac OS X,Solaris,Tru64 Unix Operating Systems Installation and Configuration Requirements Quick Reference (Doc ID 169706.1)
6. AIX 5L Practical Performance Tools and Tuning Guide Redbooks
   http://www.redbooks.ibm.com/abstracts/SG246478.html
7. Solaris Tunable Parameters Reference Manual
   http://www.sun.com/bigadmin/jsp/descFile.jsp?url=descAll/solaris_10_tunable_
8. Tuning Windows Server 2003 on IBM System x Servers Redbooks
   http://www.redbooks.ibm.com/redpapers/pdfs/redp3943.pdf
9. Performance Tuning Guidelines for Windows Server 2008, Microsoft Corporation
10. Linux Performance and Tuning Guidelines Redbooks
    http://www.redbooks.ibm.com/abstracts/REDP4285.html
11. IBM DB2 Database for Linux, UNIX, and Windows Information Center
12. IBM Content Manager for Multiplatforms Version 8.1 System Administration Certification Study Guide

Appendix B: Thanks

Thanks to the following members of the IBM DB2 Content Manager V8.5.x Enterprise Edition Performance Team for contributing to this paper.

Author of the paper:

Ang Yi; yiang@cn.ibm.com
Jian Liu; liujianl@cn.ibm.com
Zhuo Yin; yinzhuo@cn.ibm.com

Thanks to the following reviewers of the document:
Dave Royer

IBM Silicon Valley Lab, 555 Bailey Ave. San Jose, California 95141, USA.

The e-business logo, the eServer logo, IBM, the IBM logo, IBM Directory Server, DB2, Content Manager and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries or both.

The following are trademarks of other companies:
Solaris, Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries or both. Windows 2003 and Windows 2008 Enterprise Edition are trademarks of Microsoft Corporation in the United States and/or other countries. Oracle 10g and all Oracle-based trademarks and logos are trademarks of the Oracle Corporation in the United States, other countries or both. Other company, product and service names may be trademarks or service marks of others.

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PAPER .AS IS. WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you. Information in this paper as to the availability of products was believed accurate as of the time of publication. IBM cannot guarantee that identified products will continue to be made available by their suppliers. This information could include technical inaccuracies or typographical errors. Changes may be made periodically to the information herein; these changes may be incorporated in subsequent versions of the paper. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this paper at any time without notice.

Any references in this document to non-IBM web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents.
You can send license inquiries, in writing, to:
IBM Director of Licensing
IBM Corporation
North Castle Drive

Armonk, NY, USA 10504-1785