Performance and Health Monitoring

This chapter is organized as follows:

- Collected Performance Data, page 3-1
- Monitored Health Metrics, page 3-15

Collected Performance Data

This section summarizes the performance data collected by the collectd monitoring agent which is installed on all nodes. While some of the collected system-specific performance data is common for all nodes (for example disk space, CPU), the collectd agent uses plug-ins to collect application-specific data (for example for MBean, Tomcat, Apache).

This data can be accessed in several ways:

- From the Director UI > System > Stats.
- Through the WebEx Social API.

Type	Instance	Matrix	Description	Units	Expected Values	Role
Disk Usage	boot	used	Used space on partition /boot	Bytes	<99%	All
		reserved	Space on /boot partition reserved for root user.	Bytes		
		free	Free space on partition /boot	Bytes		
	opt	used	Used space on partition /opt	Bytes	<99%	All
		reserved	Space on /opt partition reserved for root user.	Bytes		
		free	Free Space on /opt partition.	Bytes		
	root	used	Used space on partition /	Bytes	<99%	
		reserved	Space on /opt partition reserved for root user.	Bytes		
		free	Free Space on /opt partition.	Bytes		

Type	Instance	Matrix	Description	Units	Expected Values	Role
Disk	sdb	disk_merged read	The number of read operations, that could be merged into other, already queued operations, i.e. one physical disk access served two or more logical operations.	Merged Operations/sec		Director-Web Message Queue, Search Store, Analytics Store, JSON Store, RDBMS Store, Index Store
		disk_merged write	The number of write operations, that could be merged into other, already queued operations, i.e. one physical disk access served two or more logical operations.	Merged Operations/sec		
		disk_octets read	Bytes read from disk per second	Bytes/sec		
		disk_octets write	Bytes written to disk per second	Bytes/sec		
		disk_ops read	Read operation from disk per seconds	Operations/sec		
		disk_ops write	Write operation to disk per seconds.	Operations/sec		
		disk_time read	Average time an I/O- read operation took to complete, equivalent to svctime of vmstat	Sec		
		disk_time write	Average time an I/O-write operation took to complete, equivalent to svctime of vmstat	Sec		
Interface	eth0	if_errors rx	Rate of Error in receiving data by network interface.	Errors/sec		All
		if_errors tx	Rate of Error in transmitting data by network interface.	Errors/sec		
		if_octets rx	Rate of Bytes received by network interface.	Bytes/sec		
		if_octets tx	Rate of Bytes transferred by network interface.	Bytes/sec		
		if_packets rx	Rate of packets receivedby network interface	Packets/sec		
		if_packets tx	Rate of packets transferred by network interface	Packets/sec		
Load		longterm	Average system load over 15 min period of time.	Average number of runnable tasks in the run-queue (15 min)		All
		midterm	Average system load over 5 min period of time.	Average number of runnable tasks in the run-queue (5 min)		
		shortterm	Average system load over 1 min period of time. Refer top/w/uptime man page for more details.	Average number of runnable tasks in the run-queue (1 min)		

Type	Instance	Matrix	Description	Units	Expected Values	Role
Swap	swap	cached	Memory that once was swapped out is swapped back in but still also is in the swapfile (if memory is needed it doesn't need to be swapped out AGAIN because it is already in the swapfile. This saves I/O) (http://www.redhat.com/advice/tips/meminfo.html/)	Bytes		All
		free	Total amount of swap space available.	Bytes		
		used	Total amount of swap space used	Bytes		
	swap_io	in	Amount of memory swapped in from disk	Kilobytes the system has swapped in from disk per second		All
		out	Amount of memory swapped out from disk	Kilobytes the system has swapped out to disk per second		
VMWare	CPU	elapsed_ms	Retrieves the number of milliseconds that have passed in the virtual machine since it last started running on the server. The count of elapsed time restarts each time the virtual machine is powered on, resumed, or migrated using VMotion.	Milliseconds		All
		limit_mhz	Retrieves the upper limit of processor use in MHz available to the virtual machine.			
		reservation_mhz	Retrieves the minimum processing power in MHz reserved for the virtual machine.			
		shares	Retrieves the number of CPU shares allocated to the virtual machine.			
		stolen_ms	Retrieves the number of milliseconds that the virtual machine was in a ready state (able to transition to a run state), but was not scheduled to run	Milliseconds		
		used_ms	Retrieves the number of milliseconds during which the virtual machine has used the CPU. This value includes the time used by the guest operating system and the time used by virtualization code for tasks for this virtual machine. Percentage of cpu utilization is used_ms*number_of_core/elapsed_ms	Milliseconds		
	Memory	active_mb	Retrieves the amount of memory the virtual machine is actively using—its estimated working set size	MegaBytes		All
		balooned_mb	Retrieves the amount of memory that has been reclaimed from this virtual machine by the vSphere memory balloon driver (also referred to as the vmmemctl driver)	MegaBytes		
		limit_mb	Retrieves the upper limit of memory that is available to the virtual machine.	MegaBytes		
		mapped_mb	Retrieves the amount of memory that is allocated to the virtual machine. Memory that is ballooned, swapped, or has never been accessed is excluded	MegaBytes		
		reservation_mb	Retrieves the minimum amount of memory that is reserved for the virtual machine	MegaBytes		
		shares	Retrieves the amount of physical memory associated with this virtual machine that is copy-on-write (COW) shared on the host.			
		swapped_mb	Retrieves the amount of memory that has been reclaimed from this virtual machine by transparently swapping guest memory to disk	MegaBytes		
		used_mb	Retrieves the estimated amount of physical host memory currently consumed for this virtual machine's physical memory	MegaBytes		

Type	Instance	Matrix	Description	Units	Expected Values	Role
Apache	apache_connections		Total number of busy workers (BusyWorkers)			App Server
	apache_idle_workers		Total number of idle workers (IdleWorkers)			
	apache_scoreboard	closing	Total number of child processes Closing connections			App Server
		dnslookup	Total number of child precesses performing DNS lookups			
		finishing	Total number of child processes Gracefully finishing			
		idle_cleanup	Total number of Idle cleanup of worker			
		keepalive	Total number of child processes maintaining KeepAlive (read) connections			
		logging	Total number of child precesses simultaneously writing to the logs			
		open	Total number of Open slot with no current process			
		reading	Total number of child processes Reading Request			
		sending	Total number of child processes Sending Reply to request			
		starting	Total number of child processes Starting up			
		waiting	Total number of child processes Waiting for Connection			
State Manager	StateManager HTTP Response Code. 200=OK, 500=ERROR	activemq-code	WxS connectivity status with Message Queue service (ActiveMQ)		200, 500	App Server
		cache-code	WxS connectivity status with Cache service		200, 500	
		digest-code	WxS connectivity status with Digest service		200, 500	
		graph-code	WxS connectivity status with Graph service		200, 500	
		index-code	WxS connectivity status with Index/Search service		200, 500	
		json-code	WxS connectivity status with JSON service		200, 500	
		notifier-code	WxS connectivity status with Notifier service		200, 500	
		quad-code	Overall connectivity status of WxS with critical services (RDBMS, JSON, Message Queue, Search, Index)		200, 500	
		quad_analytics-code	WxS connectivity status with Analytics service		200, 500	
		rabbitmq-code	WxS connectivity status with Message Queue service (RabbitMQ)		200, 500	
		rdbms-code	WxS connectivity status with RDBMS service		200, 500	
		recommendation-code	WxS connectivity status with Recommendation service		200, 500	
		search-code	WxS connectivity status with Search/Index service		200, 500	
Processes	fork	fork_rate	Number of new process forked per second.			All
	ps_state	blocked	Count of processes in Blocked state. If consistently high, alert condition need attention.			All
		paging	Count of processes in Paging state. If consistently high or growing, alert condition need attention.			
		running	Count of processes in running state. Typically less or equal to num of cores.			
		sleeping	Count of processes in sleeping state. Typically most processes are in this state.			
		stopped	Count of processes in Stopped state			
		zombies	Count of processes in Zombies state. If consistently high or growing, alert condition need attention.			

Type	Instance	Matrix	Description	Units	Expected Values	Role
TCP Connection	Port 80 - App Server, Port 61616 - Message Queue, Port 8983 - Search Store, Port 7973 - Index Store, Port 27001 - Analytics Store, Port 27000 - JSON Store, Port 11211 - Cache	close_wait	(both server and client) represents waiting for a connection termination request from the local user	number of connections		App Server, Message Queue, Search Store, Index Store, Analytics Store, JSON Store, Cache
		closed	(both server and client) represents no connection state at all	number of connections		
		closing	(both server and client) represents waiting for a connection termination request acknowledgment from the remote TCP	number of connections		
		established	(both server and client) represents an open connection, data received can be delivered to the user. The normal state for the data transfer phase of the connection	number of connections		
		fin_wait1	(both server and client) represents waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent	number of connections		
		fin_wait2	(both server and client) represents waiting for a connection termination request from the remote TCP	number of connections		
		last_ack	(both server and client) represents waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request)	number of connections		
		listen	(server) represents waiting for a connection request from any remote TCP and port	number of connections		
		syn_recv	(server) represents waiting for a confirming connection request acknowledgment after having both received and sent a connection request	number of connections		
		syn_sent	(client) represents waiting for a matching connection request after having sent a connection request	number of connections		
		time_wait	(either server or client) represents waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request. [According to RFC 793 a connection can stay in TIME-WAIT for a maximum of four minutes known as a MSL (maximum segment lifetime).]	number of connections		

Type	Instance	Matrix	Description	Units	Expected Values	Role
Oracle		blockingLock	Locks that are blocking other sessions. Should be as low as possible and should be for shorter durations.			RDBMS Store
		cacheHitRatio	Cache hit ratios should be as high as possible (highest is 100%)	%		
		dbBlockBufferCacheHitRatio	DB block buffer cache hit ratios should be as high as possible (highest is 100%)	%		
		dictionaryCacheHitRatio	Dictionary cache hit ratios should be as high as possible (highest is 100%).	%		
		diskSortRatio	Disk sorting should be minimal			
		invalidObjects	Invalid objects should be as minimal as possible			
		latchHitRatio	Latch hit ratios should be as high as possible (highest is 100%)	%		
		libraryCacheHitRatio	Library Cache hit ratios should be as high as possible (highest is 100%)	%		
		lock	Minimum number of locks for shorter durations			
		lockedUserCount	The QUADDB and XMPP accounts should be unlocked and so are the DBA/other accounts such as SYS, SYSTEM, SYSMAN etc.			
		offlineDataFiles	All the Datafiles should be ONLINE			
		pgaInMemorySortRatio	PGA memory sorts should be as high as possible			
		rollBlockContentionRatio	Should be minimal			
		rollHeaderContentionRatio	Should be minimal			
		rollHitRatio	Should be as high as possible			
		rollbackSegmentWait	Should be minimal			
		sessionPGAMemory	PGA memory consumed by a session			
		sessionUGAMemory	UGA memory consumed by a session			
		sgaDataBufferHistRatio	Hit ratios should be as high as possible (highest is 100%)	%		
		sgaSharedPoolFree	Too much of free shared pool means over allocation/wastage of memory resource. No shared pool being free can be an indication of memory starving.			
		sgaSharedPoolReloadRatio	System Global Area shared pool reload ratio			
		softParseRatio	Soft parse ratio of the SQLs			
		staleStatistics	Statistics should be up-to-date			
	ioPerTableSpace: ecp_data, sysaux, system, undotbs1, users	PHY_BLK_R	Physical Blocks Read			RDBMS Store, Graph Store
		Phy_BLK_W	Physical Blocks WRITE			
	oraUsageTablespace: ecp_data, sysaux, system, undotbs1, users	free_mb	Free Space in MB	MegaBytes		RDBMS Store, Graph Store
		percent_free	% Free Space	%		
		percent_used	% Used	%		
		size_mb	Size in MB	MegaBytes		

Type	Instance	Matrix	Description	Units	Expected Values	Role
Solr	Search	avgRequestsPerSecond	Number of requests server per second	Seconds		Search Store
		avgTimePerRequest	average time taken to server each request	Milliseconds		
		errors	Rate of error, requests that returned error.	Number		
		requests	Rate of request servered by SOLR.	Number		
		timeouts	Rate of request timed out, request that failed due to time out error.	Number		
	Search: documentcache, fieldvaluecache, filtercache, queryresultcache Index: autocompletefieldvalue, followerfieldvaluecach, postfieldvaluecache, socialfieldvaluecache, videofieldvaluecache	cumulative_evictions	The number of entries that have been removed from the cache, from the start of the solr server	Number		Search Store, Index Store
		cumulative_hits	This number denotes the total number of lookups that were sent to the cache that resulted in positive match in the cache, from the start of the solr server	Number		
		cumulative_inserts	The total number of values inserted in the cache, from the start of the solr server	Number		
		cumulative_lookups	This number shows the total number of lookups/reads on the cache from the start of the solr server	Number		
		evictions	The number of entries that have been removed from the cache	Number		
		hitratio	The percentage of accesses that result in cache hits is known as the hit rate or hit ratio of the cache	Number		
		hits	The number of documents returned upon search	Number		
		inserts	The number of entries that have been added to the cache	Number		
		lookups	The number of lookups/reads on the cache, since the last cache invalidation (or last commit operation)	Number		
		size	Maximum number of entries in the cache	Number		
		warmupTime	Time to warm up the cache in milliseconds.	Milliseconds		
	Search: searcher Index: autocomplete, follower, post, social, video	maxDoc	maxDoc is the maximum internal document id currently in use. The difference between maxDocs and numDocs numbers gives an idea of how many “deleted” (or replaced) documents are currently still in the index. They gradually get cleaned up as segments get merged or when the index gets optimized.	Number		Search Store, Index Store
		numDocs	numDocs is the number of unique “live” Documents in the solr index. It’s how many docs you would get back from a query for *:*.	Number		

Type	Instance	Matrix	Description	Units	Expected Values	Role
Java Memory	HeapMemoryUsage: Current memory usage of the heap that is used for object allocation. The heap consists of one or more memory pools. The used and committed size of the returned memory usage is the sum of those values of all heap memory pools whereas the init and max size of the returned memory usage represents the setting of the heap memory which may not be the sum of those of all heap memory pools. The amount of used memory in the returned memory usage is the amount of memory occupied by both live objects and garbage objects that have not been collected, if any. NonHeapMemoryUsage: Current memory usage of non-heap memory that is used by the Java virtual machine. The non-heap memory consists of one or more memory pools. The used and committed size of the returned memory usage is the sum of those values of all non-heap memory pools whereas the init and max size of the returned memory usage represents the setting of the non-heap memory which may not be the sum of those of all non-heap memory pools.	HeapMemoryUsage_committed	Represents the amount of memory (in bytes) that is guaranteed to be available for use by the Java virtual machine. The amount of committed memory may change over time (increase or decrease). The Java virtual machine may release memory to the system and committed could be less than init.committed will always be greater than or equal to used.	Bytes		Search Store, Index Store, Message Queue, App Server, Worker
		HeapMemoryUsage_init	Represents the initial amount of memory (in bytes) that the Java virtual machine requests from the operating system for memory management during startup. The Java virtual machine may request additional memory from the operating system and may also release memory to the system over time. The value of init may be undefined.	Bytes		
		HeapMemoryUsage_max	Represents the maximum amount of memory (in bytes) that can be used for memory management. Its value may be undefined. The maximum amount of memory may change over time if defined. The amount of used and committed memory will always be less than or equal to max if max is defined. A memory allocation may fail if it attempts to increase the used memory such that used > committed even if used <= max would still be true (for example, when the system is low on virtual memory).	Bytes		
		HeapMemoryUsage_used	Represents the amount of memory currently used (in bytes).	Bytes		
		NonHeapMemoryUsage_committed	Represents the amount of memory (in bytes) that is guaranteed to be available for use by the Java virtual machine. The amount of committed memory may change over time (increase or decrease). The Java virtual machine may release memory to the system and committed could be less than init.committed will always be greater than or equal to used.	Bytes		
		NonHeapMemoryUsage_init	Represents the initial amount of memory (in bytes) that the Java virtual machine requests from the operating system for memory management during startup. The Java virtual machine may request additional memory from the operating system and may also release memory to the system over time. The value of init may be undefine.	Bytes		
		NonHeapMemoryUsage_max	Represents the maximum amount of memory (in bytes) that can be used for memory management. Its value may be undefined. The maximum amount of memory may change over time if defined. The amount of used and committed memory will always be less than or equal to max if max is defined. A memory allocation may fail if it attempts to increase the used memory such that used > committed even if used <= max would still be true (for example, when the system is low on virtual memory).	Bytes		
		NonHeapMemoryUsage_used	Represents the amount of memory currently used (in bytes).	Bytes		
Java fd		OpenFileDescriptorCount	Number of all file handles taken by the Java virtual machine currently. This includes all created sockets and virtual machine resources, too. Example notification value: (MaxFileDescriptorCount - OpenFileDescriptorCount) < 100. Monitor to determine if the number of open files that can be opened by the vm is sufficient.			Search Store, Index Store

Type	Instance	Matrix	Description	Units	Expected Values	Role
Non Java Application processes	ps_count	processes	Total number of processes (including child) forked for particular program.			Analytics Store, JSON Store, Cache, RabbitMQ
		threads	Total number of threads created for particular program.			
	ps_code		Total (in KB) of Shared library code size (VmLib) & Size of text segment (VmExe)	KiloBytes		Analytics Store, JSON Store, Cache
	ps_data		Size (in KB) of data segment (VmData)	KiloBytes		Analytics Store, JSON Store, Cache
	ps_rss		Number of pages the process has in real memory. This is just the pages which count towards text, data, or stack space. This does not include pages which have not been demand-loaded in, or which are swapped out.			Analytics Store, JSON Store, Cache
	ps_stacksize		Stack size. Difference between the address of the start of the stack (startstck) & current value of ESP stack pointer, as found in the kernel stack page for the process (kstkesp).			Analytics Store, JSON Store, Cache
	ps_vm		Virtual memory size in bytes.	Bytes		Analytics Store, JSON Store, Cache
	ps_cputime	syst	Amount of time that this process has been scheduled in kernel mode, measured in clock ticks (divide by sysconf(_SC_CLK_TCK)).			Analytics Store, JSON Store, Cache
		user	Amount of time that this process has been scheduled in user mode, measured in clock ticks (divide by sysconf(_SC_CLK_TCK)). This includes guest time, guest_time (time spent running a virtual CPU), so that applications that are not aware of the guest time field do not lose that time from their calculations.			
	ps_disk_octets	read	I/O counter: chars read The number of bytes which this task has caused to be read from storage. This is simply the sum of bytes which this process passed to read() and pread(). It includes things like tty IO and it is unaffected by whether or not actual physical disk IO was required (the read might have been satisfied from pagecache).			Analytics Store, JSON Store, Cache
		write	I/O counter: chars written The number of bytes which this task has caused, or shall cause to be written to disk. Similar caveats apply here as with rchar.			
	ps_disk_ops	read	I/O counter: read syscalls Attempt to count the number of read I/O operations, i.e. syscalls like read() and pread().			Analytics Store, JSON Store, Cache
		write	I/O counter: write syscalls Attempt to count the number of write I/O operations, i.e. syscalls like write() and pwrite().			
	ps_pagefaults	majfit	The number of major faults the process has made which have required loading a memory page from disk.			Analytics Store, JSON Store, Cache
		minfit	The number of minor faults the process has made which have not required loading a memory page from disk.			

Type	Instance	Matrix	Description	Units	Expected Values	Role
MongoDB		cache_misses	'serverStatus.indexCounters.accesses' divided by 'serverStatus.indexCounters.misses' serverStatus.indexCounters.accesses: accesses reports the number of times that operations have accessed indexes. This value is the combination of the hits and misses. Higher values indicate that your database has indexes and that queries are taking advantage of these indexes. If this number does not grow over time, this might indicate that your indexes do not effectively support your use. serverStatus.indexCounters.misses: misses represents the number of times that an operation attempted to access an index that was not in memory. These “misses,” do not indicate a failed query or operation, but rather an inefficient use of the index. Lower values in this field indicate better index use and likely overall performance as well			Analytics Store, JSON Store
		connections	serverStatus.connections.current: The value of current corresponds to the number of connections to the database server from clients. This number includes the current shell session. Consider the value of available to add more context to this datum. This figure will include the current shell connection as well as any inter-node connections to support a replica set or sharded cluster.			
		page_fault	serverStatus.extra_info.page_faults:Reports the total number of page faults that require disk operations. Page faults refer to operations that require the database server to access data which isn’t available in active memory. The page_faults counter may increase dramatically during moments of poor performance and may correlate with limited memory environments and larger data sets. Limited and sporadic page faults do not necessarily indicate an issue.			
		lock_ratio%	Displays the relationship between lockTime and totalTime. Low values indicate that operations have held the globalLock frequently for shorter periods of time. High values indicate that operations have held globalLock infrequently for longer periods of time serverStatus.globalLock.totalTime: The value of totalTime represents the time, in microseconds, since the database last started and creation of the globalLock. This is roughly equivalent to total server uptime. serverStatus.globalLock.lockTime: The value of lockTime represents the time, in microseconds, since the database last started, that the globalLock has been held. Consider this value in combination with the value of totalTime. MongoDB aggregates these values in the ratio value. If the ratio value is small but totalTime is high the globalLock has typically been held frequently for shorter periods of time, which may be indicative of a more normal use pattern. If the lockTime is higher and the totalTime is smaller (relatively,) then fewer operations are responsible for a greater portion of server’s use (relatively.)			
	flushes	flushes	serverStatus.backgroundFlushing.flushes: flushes is a counter that collects the number of times the database has flushed all writes to disk. This value will grow as database runs for longer periods of time.			
		flushes_avg_ms	serverStatus.backgroundFlushing.average_ms:The average_ms value describes the relationship between the number of flushes and the total amount of time that the database has spent writing data to disk. The larger flushes is, the more likely this value is likely to represent a “normal,” time; however, abnormal data can skew this value. Use the last_ms to ensure that a high average is not skewed by transient historical issue or a random write distribution.			

Type	Instance	Matrix	Description	Units	Expected Values	Role
	memory	mapped	serverStatus.mem.mapped: The value of mapped provides the amount of mapped memory, in megabytes (MB), by the database. Because MongoDB uses memory-mapped files, this value is likely to be to be roughly equivalent to the total size of your database or databases.	MegaBytes		
		resident	serverStatus.mem.resident: The value of resident is roughly equivalent to the amount of RAM, in megabytes (MB), currently used by the database process. In normal use this value tends to grow. In dedicated database servers this number tends to approach the total amount of system memory.	MegaBytes		
		virtual	serverStatus.mem.virtual: virtual displays the quantity, in megabytes (MB), of virtual memory used by the mongod process. In typical deployments this value is slight ly larger than mapped. If this value is significantly (i.e. gigabytes) larger than mapped, this could indicate a memory leak. With journaling enabled, the value of virtual is twice the value of mapped.	MegaBytes		
	network	bytesin	serverStatus.network.bytesIn: The value of the bytesIn field reflects the amount of network traffic, in bytes, received by this database. Use this value to ensure that network traffic sent to the mongod process is consistent with expectations and overall inter-application traffic.	Bytes		
		bytesout	serverStatus.network.bytesOut: The value of the bytesOut field reflects the amount of network traffic, in bytes, sent from this database. Use this value to ensure that network traffic sent by the mongod process is consistent with expectations and overall inter-application traffic.	Bytes		
	oplogs	difftimesec	Time difference between the most recent and the oldest oplog.			
		storagesizemb	The total amount of storage (in MB) allocated to this collection for document storage. The storageSize does not decrease as you remove or shrink documents.	MegaBytes		
		usedsizemb	The size (in MB) of the data stored in this collection. This value does not include the size of any indexes associated with the collection.	MegaBytes		

Type	Instance	Matrix	Description	Units	Expected Values	Role
	replication	health	The health value is only present for the other members of the replica set. This field conveys if the member is up (i.e. 1) or down (i.e. 0.)		Up=1, Down=0	
		optimelagsec	Replication lag between secondary node and primary node			
		state	The value of the state reflects state of this replica set member.		An integer between 0 and 10 represents the state of the member. These integers map to states, as follows: 0 STARTUP Startup, phase 1 (parsing config.) 1 PRIMARY Primary. 2 SECONDARY Secondary. 3 RECOVERING Member is recovering (initial sync, post-rollback, stale members.) 4 FATAL Member has encountered unrecoverable error. 5 STARTUP2 Start up, phase 2 (forking threads.) 6 UNKNOWN Unknown (the set has never connected to the member.) 7 ARBITER Member is an arbiter. 8 DOWN Member is not accessible to the set. 9 ROLLBACK Member is rolling back data. 10 SHUNNED Member has been removed from replica set.	

Type	Instance	Matrix	Description	Units	Expected Values	Role
	total_operations Note: The opcounters data structure provides an overview of database operations by type and makes it possible to analyze the load on the database in more granular manner. These numbers will grow over time and in response to database use. Analyze these values over time to track database utilization.	command	Provides a counter of the total number of commands issued to the database since the mongod instance last started			
		delete	Provides a counter of the total number of delete operations since the mongod instance last started			
		getmore	Provides a counter of the total number of “getmore” operations since the mongod instance last started. This counter can be high even if the query count is low. Secondary nodes send getMore operations as part of the replication process			
		insert	Provides a counter of the total number of insert operations since the mongod instance last started			
		query	Provides a counter of the total number of queries since the mongod instance last started			
		update	Provides a counter of the total number of update operations since the mongod instance last started			
MongoDB databases	quad, recommendation	collections	Contains a count of the number of collections in that database			
		indexes	Contains a count of the total number of indexes across all collections in the database			
		num_extents	Contains a count of the number of extents in the database across all collections			
		object_count	Contains a count of the number of objects (i.e. documents) in the database across all collections			
		data file_size	The total size of the data held in this database including the padding factor. The dataSize will not decrease when documents shrink, but will decrease when you remove documents	Bytes		
		index file_size	The total size of all indexes created on this database	Bytes		
		storage file_size	The total amount of space allocated to collections in this database for document storage. The storageSize does not decrease as you remove or shrink documents	Bytes		
Tomcat		activeSessions	Number of active sessions at this moment			App Server
		expiredSessions	Number of sessions that expired (doesn't include explicit invalidations)			
		processExpiresFrequency	The frequency of the manager checks (expiration and passivation)			
		processingTime	Time spent doing housekeeping and expiration	Cumulative milliseconds of wall clock elapsed time		
		rejectedSessions	Number of sessions rejected due to maxActive being reached			
		sessionAverageAliveTimes	Average time an expired session had been alive	Seconds		
		sessionCounter	Total number of sessions created by this manager			
		sessionCreateRate	Session creation rate in sessions per minute	Minute		
		sessionExpireRate	Session expiration rate in sessions per minute	Minute		

Type	Instance	Matrix	Description	Units	Expected Values	Role
RabbitMQ	Queue: Activity, Analytics, EMailDigest, Migrate, Polling, Scheduler	consumers	Number of consumers for the queue			Message Queue
		memory	Bytes of memory consumed by the Erlang process associated with the queue, including stack, heap and internal structures.	Bytes		
		messages	Sum of ready and unacknowledged messages (queue depth).			
		messages_ready	Number of messages ready to be delivered to clients.			
		messages_acknowledged	Number of messages delivered to clients but not yet acknowledged.			
		node	Node associated with the queue			
	Server	fd_total	File descriptor count and limit, as reported by the operating system. The count includes network sockets and file handles.			Message Queue
		fd_used	File descriptor count used by RabbitMQ.			
		mem_limit	The memory threshold RabbitMQ will use on the system.	Bytes		
		mem_used	Memory used by RabbitMQ	Bytes		
		proc_total	Maximum number of erlang processes for RabbitMQ			
		proc_used	Number of erlang processes used by RabbitMQ			
		sockets_total	The network sockets count and limit managed by RabbitMQ.			
		sockets_used	The network sockets count used by RabbitMQ.			
		uptime	Uptime of the service	Milliseconds		
ActiveMQ	Broker	TotalEnqueueCount	Number of messages sent to queues			Message Queue
		TotalDequeueCount	Number of messages removed from queues & consumed by the clients			
		TotalConsumerCount	Number of clients listening to the queue			
		TotalMessageCount	Number of Messages held by the broker. [TotalMessagesCount+TotalDequeueCount = TotalEnqueueCount]			
		MemoryLimit	The memory usage limit of the broker	Bytes		
		MemoryPercentUsage	Percentage usage of the memory	%		
		StoreLimit	The upper limit of the store usage of the broker -- we haven't configured any upper limit for WxS queues			
		StorePercentUsage	The actual storage usage of the broker			

Type	Instance	Matrix	Description	Units	Expected Values	Role
ActiveMQ	Queue: inbound, outbound, portal, search, vdl	QueueSize	Total number of messages in the queue/store that have not been ack'd by a consumer			Message Queue
		EnqueueCount	Total number of messages sent to consumer sessions (Dequeue + Inflight)			
		DequeueCount	Number of messages sent to a consumer session and have not received an ack			
		ConsumerCount	Total number of messages sent to the queue since the last restart			
		DispatchCount	Total number of messages removed from the queue (ack'd by consumer) since last restart			
		ExpiredCount	Number of client/consumers listening on this Queue			
		InFlightCount	Number of messages which didn't get sent to the clients/Consumers and reach the expiry timeout and cleared by broker -- We have the expired timeout of 8 hours			
		CursorMemory Usage	Indicates the memory(heap) used by non-persistent messages -- this doesn't to WxSocial as we use persistent messaging			
		CursorPercentU sage	Indicates the memory(heap) used by non-persistent messages in percentage	%		
		MemoryLimit	The upper limit of memory usage of a particular Queue—WxS we haven't configured any upper limits for the Queues in WxS	Bytes		
		MemoryPercent Usage	The percentage of memory usage of a particular Queue	%		

Monitored Health Metrics

This section summarizes the resources that are monitored by monit to ensure good health of the system. Monit automatically takes corrective action if a process stops or becomes unresponsive. A syslog message is generated on alert and when corrective action is taken. Monit checks are only done on Enabled applications.

This data can be accessed in several ways:

- From the Director UI > System > Health.
- Through the WebEx Social API.

Table 3-1 Monitored Health Metrics

CheckName/ Filename	Type	Checks	Action	Role
jms-message-queue/ process_activemq	Process	pid	Restart	Message Queue
		cpu > 98% for 5 polls	Syslog Err Msg	
analyticsstore/ process_analyticsstore	Process	pid	Restart	Analytic Store
		tcp on port 27001 for 1 poll	Syslog Err Msg	
analyticsstore/ process_analyticsstore[1]	Process	pid	Restart	Director
		tcp on port 27001 for 1 poll	Syslog Err Msg	
		cpu > 98% for 5 polls	Syslog Err Msg	

Table 3-1 Monitored Health Metrics (continued)

CheckName/ Filename	Type	Checks	Action	Role
cache/ process_cache	Process	pid	Restart	Cache
		Built-in monit protocol check for memcache on port 11211 for 1 poll	Syslog Err Msg	
		cpu > 98% for 5 polls	Syslog Err Msg	
carbon/ process_carbon	Process	pid	Restart	Director
		cpu > 25% for 5 polls	Syslog Err Msg	
cmanager/ process_cmanager	Process	pid	Restart	WebEx Social
		cpu > 98% for 5 polls	Syslog Err Msg	
collectd/ process_collectd	Process	pid	Restart	All
		cpu > 25% for 5 polls	Syslog Err Msg	
director-web/ process_cps	Process	pid	Restart	Director
		cpu > 98% for 5 polls	Syslog Err Msg	
	Disk Space	/opt > 85% for 5 polls	Purge /opt/logs/*, except for today's log	
cron/ process_cron	Process	pid	Restart	All
httpd/ process_httpd	Process	pid	Restart	Director, WebEx Social, Worker
indexstore/ process_indexstore	Process	pid	Restart	Index Store
		cpu > 98% for 5 polls	Syslog Err Msg	
jsonstore/ process_jsonstore	Process	pid	Restart	JSON Store
		tcp on port 27000 for 1 poll	Syslog Err Msg	
		cpu > 98% for 5 polls	Syslog Err Msg	
jsonstore/ process_jsonstore[1]	Process	pid	Restart	Director
		tcp on port 27000 for 1 poll	Syslog Err Msg	
		cpu > 98% for 5 polls	Syslog Err Msg	
nagios/ process_nagios	Process	pid	Restart	Director
		cpu > 25% for 5 polls	Syslog Err Msg	
ntpd/ process_ntpd	Process	pid	Restart	All
		cpu > 25% for 5 polls	Syslog Err Msg	
notifier/ process_openfire	Process	pid	Restart	Notifier
		cpu > 98% for 5 polls	Syslog Err Msg	
postfix/ process_postfix[2]	Process	pid	Restart	Director, Worker
		cpu > 40% for 2 polls	Syslog Err Msg	
		cpu > 60% for 5 polls	Restart	
		Built-in monit protocol check for SMTP for 1 poll	Syslog Err Msg	
		Children > 2000	Syslog Err Msg	
		Memory > 2GB for 2 polls	Restart	
puppet/ process_puppet	Process	pid	Restart	All
		cpu > 98% for 5 polls	Syslog Err Msg	
puppetmaster/ process_puppetmaster	Process	pid	Restart	Director
		tcp on port 8140 for 1 poll	Syslog Err Msg	
		cpu > 98% for 5 polls	Syslog Err Msg	

Table 3-1 Monitored Health Metrics (continued)

CheckName/ Filename	Type	Checks	Action	Role
quad/ process_quad	Process	pid	Restart	WebEx Social
		cpu > 98% for 5 polls	Syslog Err Msg	
		WxS State Manager URL check for 2 polls[3]	Syslog Err Msg	
message-queue/ process_rabbitmq	Process	pid	Restart	Message Queue
		cpu > 98% for 5 polls	Syslog Err Msg	
rsyslog/ process_rsyslog	Process	pid	Restart	All
		tcp on port 514 for 1 poll	Syslog Err Msg	Director
		cpu > 50% for 5 polls	Syslog Err Msg	All
saltmaster/ process_saltmaster	Process	pid	Restart	Director
		tcp on port 4506 for 1 poll	Syslog Err Msg	
		cpu > 98% for 5 polls	Syslog Err Msg	
saltminion/ process_saltminion	Process	pid	Restart	All
		cpu > 98% for 5 polls	Syslog Err Msg	
search/ process_searchstore	Process	pid	Restart	Search Store
		cpu > 98% for 5 polls	Syslog Err Msg	
sshd/ process_sshd	Process	pid	Restart	All
		Built-in monit protocol check for ssh on port 22 for 1 poll	Syslog Err Msg	
		cpu > 25% for 5 polls	Syslog Err Msg	
worker/ process_worker	Process	pid	Restart	Worker
		cpu > 98% for 5 polls	Syslog Err Msg	
oracle/ program_oracle[4]	Program (script)	script return value; for 10 polls	Restart	RDBMS Store, Graph Store
integrity/ program_integrity	Program (script)	script return value;	Syslog Err Msg	All
Disk usage check[5]	/opt	> 85%	Nagios Alert	All Note: Nagios Alert when /opt usage > 85% is for the Director role only.
	/opt	> 95%	Nagios Alert	
	/boot	> 99%	Nagios Alert	
	/root	> 99%	Nagios Alert	
Filesystems[3]	/opt. /boot, /root & NFS (where mounted)	Not Writable for 2 polls	Nagios Alert	All

1. Arbiter check available only where there are multiple Json/Analytics VMs.
2. Postfix service monitored only when maildomain/external host and external SMTP port are provisioned.
3. Introduced in 3.3(1).
4. The check is done using "/etc/init.d/dbora status". Restarting is done using "/etc/init.d/dbora cond_start". Only services that are not running (Enterprise Manager, Database etc) are started. Checks are not made during database installation.
5. The disk utilization check uses performance statistics as collected by collectd.