

Parameter	Description
<b>Broker Application Class (DM_BROKER)</b>	
Count	graphs the number of concurrent DocBroker processes running on the Documentum server. At this time, PATROL for Documentum supports only one DocBroker per host. The collector (DBCollector) retrieves data from the PATROL for Unix PROCPPCount parameter under PROCESS:PROCESS_PRESENCE:dm-docbroker.
docbrokerMEM	graphs the total amount of virtual memory used by the DocBroker processes running on the server. At this time, PATROL for Documentum supports only one DocBroker per host. The collector (DBCollector) retrieves data from the PATROL for Unix PROCPPMem parameter under PROCESS:PROCESS_PRESENCE:dm-docbroker.
docbrokerCPU	graphs the percentage of CPU used by the DocBroker processes running on the server. At this time, PATROL for Documentum supports only one DocBroker per host. The collector (DBCollector) retrieves data from the PATROL for Unix PROCPPCPUPerc parameter under PROCESS:PROCESS_PRESENCE:dm-docbroker.
Status	reports the whether the DocBroker is responding. The parameter is set to 0 (OK) if it is responding.
<b>Collectors Application Class (DM_COLLECTORS)</b>	
brokerCollector	performs Docbroker collection for Docbroker processes running in the eContent Server host for the DM_BROKER application class. The parameter is set to 0 (OK) if it is functioning normally and 1 if it is in alarm.

Parameter	Description
DBCcollector	executes the <i>dmqdocbroker</i> utility with the <i>getservermap</i> option to retrieve parameter information for the DM_DOCBASE parameters. The parameter is set to 0 (OK) if it is functioning normally and 1 if it is in alarm.
docbaseCollector	performs Docbase collection for all aspects of a Docbase for the DM_DOCBASE, DM_FILESTORE, DM_JOBS, DM_PROCESS, and DM_USERS application classes. The parameter is set to 0 (OK) if it is functioning normally and 1 if it is in alarm.

#### Database Application Class (DM\_DATABASE)

DBConnectStatus	reports the status of the connection to database (0 (OK) if database is responding).
Ddl_ops	reports the number of data definition operations performed. Data definition operations encompass operations such as "create table" or "drop table."
Deletes	reports the number of SQL delete operations performed by the database.
Inserts	reports the number of SQL insert operations performed by the database.
Max_cursors	reports the maximum number of concurrently open cursors. A cursor is a handle for a private SQL area an area in memory in which a parsed statement and other information for processing the statement are kept. Each user session can open multiple cursors up to the limit set in the database.
Rpc_ops	reports the number of RPC (Remote Procedure Call) calls between eContent Server and the RDBMS.
Selects	reports the number of SQL select operations performed by the database.
Updates	reports the number of SQL update operations performed by the database.

#### Docbase Application Class (DM\_DOCBASE)

KeepEntry	<p>each DocBroker is told how long it can keep a Docbase entry if it does not receive checkpoint broadcasts from the Docbase. This time limit is defined by Documentum as the <i>keep_entry_interval</i>; by default the <i>keep_entry_interval</i> is set to 24 hours (expressed in minutes).</p> <p>The KeepEntry parameter is computed as:  <math>(\text{CheckPoint} \div \text{keep\_entry\_interval}) * 100</math></p> <p>When the value of this parameter approaches 100%, this indicates that the DocBroker will soon remove the Docbase entry.</p>
-----------	--

Parameter	Description
CheckPoint	graphs the time that the target Docbase last sent information to its DocBroker. A DocBroker periodically receives information from its associated Docbases. CheckPoint tracks the elapsed time since the event last occurred.
Status	this parameter is set to 0 (OK) if the DocBroker responded with a Server Status: Open to the query.

#### Filestore Application Class (DM\_FILESTORE)

FSAvailableSpace	graphs the available space for the file system. If the file system is being monitored by the PATROL for Unix product, the data is retrieved from the Unix KM. More frequently the collector (FS2Collector) executes Unix commands to collect this data.
FSCapacity	graphs the percentage of space used by the file system. This value is calculated as $(FSUsedSpace \div FSAvailableSpace) * 100$
FSUsedSpace	graphs the space used by the file system. If the file system is being monitored by the PATROL for Unix product, the data is retrieved from the Unix KM. More frequently the collector (FS2Collector) executes Unix commands to collect this data.

#### Filestore Container Application Class (DM\_FILESTORE\_CONT)

FSCollStatus	reports the status of the internal file store collector that is included in the docbaseCollector.
--------------	---

#### Jobs Application Class (DM\_JOBS)

AbortedJobs	reports the number of aborted jobs that were returned from a query requesting job status information from the Documentum server
CompletedJobs	reports the number of completed jobs that were returned from a query requesting job status information from the Documentum server
FailedJobs	reports the number of failed jobs that were returned from a query requesting job status information from the Documentum server
JobCollStatus	reports the status of the internal job collector that is included in the docbaseCollector.

#### Server Performance Application Class (DM\_PERF)

CheckInTime	reports the time it took to perform a document check-in operation against the Docbase from the client host
CheckOutTime	reports the time it took to perform a document check-out operation against the Docbase from the client host

Parameter	Description
ConnectionStatus	this status parameter is set to 100 (OK) if no errors occurred while the client program was connected to the Docbase; otherwise, it is set to 0 and goes into alarm
ConnectTime	reports the time it took for the Desktop Client Host (DM_Perf.exe) to connect to the Docbase and receive a response
CurrentSysObjCntQueryTime	reports the time it took to gather a count of system objects in the Documentum Docbase (select all counts from dm_sysobject)
CurrentSysObjects	contains the current number of system objects in the Docbase
DeleteTime	reports the time it took to perform a document delete operation against the Docbase from the client host
DisconnectTime	reports the time it took for the Desktop Client Host (DM_Perf.exe) to disconnect from the Docbase
DMPerfCollector	collector parameter of DM_PERF class. DMPerfCollector executes client program dcm_perf.exe to collect performance data from the DM_CLIENT KM. Performance is measured by sending two documents from a remote machine to a Docbase for processing, measuring the time it takes, and recording any errors that may have occurred. Communication with the client agent is done via SNMP. See also "Parameter Summary by Collector Class" on page 6-8.
DrWatsonCheck	indicates whether a Windows NT 'Dr. Watson' error occurred while executing the DM_Dbkr_Info.exe program. This is a rare occurrence.
GetPerfTime	reports the total time to collect all Docbase performance data
ImportTime	reports the time it took to perform a document import operation against the Docbase from the client host
QueryAndProcessCount	contains the number of valid_type objects processed from a returned query sent by the client host (select name from dm_format order by name)
QueryAndProcessTime	reports the time it took to query and process valid_type objects (select name from dm_format order by name)
TotalSysObjCntQueryTime	reports the time it took to gather a count of all system objects in the Documentum Docbase (select all counts from dm_sysobject(all))
TotalSysObjects	contains the number of system objects in the Docbase
<b>Client Performance Application Class (DM_PERF_CLIENT)</b>	
DrWatsonCheck	indicates whether a Windows NT "Dr. Watson" error occurred while executing the DM_Dbkr_Info.exe program. This is a rare occurrence.

Parameter	Description
DocBrokerCollector	collector parameter of DM_PERF_CLIENT class. DocBrokerCollector collects data remotely from the DM_CLIENT KM. DocBrokerCollector executes client program dctm_perf.exe. Performance is measured by sending two documents from a remote machine to a Docbase for processing, measuring the time it takes, and recording any errors that may have occurred. Communication with the server agent is done via SNMP. See also "Parameter Summary by Collector Class" on page 6-8.
DocBrokerInfoTime	reports the time it took to execute the DM_Dbkr_Info.exe program to gather information about the DocBroker running on the server, from that remote client machine
DocBaseList	displays a list that shows the Docbase that the client component of the PATROL for Documentum host has been configured to use for end-user performance data acquisition
IgnoredDocBaseList	contains a list of Docbases that are being ignored by the performance client because an error occurred while the client attempted to connect to the Docbase
<b>Process Application Class (DM_PROCESS)</b>	
Count	contains the number of concurrent 'dm_agent_exec' or 'documentum' processes running on the Documentum server
CPUUtilization	reports total CPU consumption of all concurrent 'dm_agent_exec' or 'documentum' processes running on the Documentum server
MemoryUsage	reports the total memory usage of all concurrent 'dm_agent_exec' or 'documentum' processes running on the Documentum server
ProcCollStatus	reports the status of the internal process collector that is included in the docbaseCollector.
Status	reports the whether the DocBroker is responding. The parameter is set to 1 (OK) if it is.
<b>Process Container Application Class (DM_PROC_CONT)</b>	
ProcCollStatus	reports the status of the internal process collector that is included in the docbaseCollector.
<b>Users Application Class (DM_USERS)</b>	
ActiveUsers	graphs the number of active users in the Docbase (labeled as Active in the query return string)
Idle_1	graphs the number of users that have been inactive between 1 and 2 hours
Idle_2	graphs the number of users that have been inactive between 2 and 3 hours

<b>Parameter</b>	<b>Description</b>
Idle_3	graphs the number of users that have been inactive between 3 and 4 hours
Idle_4	graphs the number of users that have been inactive between 4 and 5 hours
Idle_5	graphs the number of users that have been inactive between 5 and 6 hours
Idle_6	graphs the number of users that have been inactive between 6 and 7 hours
Idle_7	graphs the number of users that have been inactive between 7 and 8 hours
Idle_8	graphs the number of users that have been inactive longer than 8 hours
InactiveUsers	graphs the number of inactive users in the Docbase (labeled as Inactive in the query return string). Inactive sessions are sessions where the server connection has timed out but the client has not specifically disconnected from the server.
TotalUsers	graphs the total number of Documentum users, active and inactive
UserCollStatus	reports the status of the internal user collector that is included in the docbaseCollector.