Measurements

On this page

Retrieves measurements collected by the Monitoring and Automation Agents for your MongoDB processes, databases, and hardware disks. Monitoring Agents collect process and database measurements using MongoDB diagnostic commands, including serverStatus and dbStats. Automation Agents collect measurements for servers that run managed `mongod` and `mongos` processes.

Query Parameters

When you issue a `GET` command to retrieve measurements through the `measurements` endpoint, you must include the `granularity` parameter and either the `period` or `start` and `end` parameters.

Each endpoint supports the following query parameters:

Parameter	Description
`granularity`	An ISO-8601-formatted time period that specifies the interval between measurement data points. For example, `PT30S` specifies 30-second granularity. The supported values for this parameter are the same as are available in the **Granularity** drop-down list when you view metrics in the Ops Manager interface.
`period`	How far back in the past to retrieve measurements, as specified by an ISO-8601 period string. For example, setting `PT24H` specifies 24 hours. An ISO-8601-formatted time period that specifies how far back in the past to query. For example, to request the last 36 hours, specify: `period=P1DT12H`.
`start`	The time at which to start retrieving measurements, as specified by an ISO-8601 timestamp string. If you specify `start` you must also specify `end`.
`end`	The time at which to stop retrieving measurements, as specified by an ISO-8601 timestamp string. If you specify `end` you must also specify `start`.

Parameter	Description
`m`	Specifies which measurements to return. If `m` is not specified, all measurements are returned. To specify multiple values for `m`, you must repeat the `m` parameter. For example: `../measurements?m=CONNECTIONS&m=OPCOUNTER_CMD&m=OPCOUNTER_QUERY` You must specify measurements that are valid for the host. Ops Manager returns an error if any specified measurements are invalid For available measurements, see Measurement Types.

Endpoints

Get Host, Process, and System Measurements

You must include the `granularity` parameter and either the `period` or `start` and `end` parameters.

```
GET /api/public/v1.0/groups/GROUP-ID/hosts/HOST-ID/measurements?granularity=ISO-8601-P
```

Host measurements provide data on the state of the MongoDB process. The Monitoring Agent collects host measurements through the MongoDB serverStatus and dbStats commands.

System and process measurements provide data on the CPU usage of the servers that run MongoDB. The Automation Agent collects these measurements. You must run Ops Manager Automation to retrieve system and process measurements.

Get Disk Partition Measurements

You must include the `granularity` parameter and either the `period` or `start` and `end` parameters.

```
GET /api/public/v1.0/groups/GROUP-ID/hosts/HOST-ID/disks/PARTITION-NAME/measurements
```

Disk measurements provide data on IOPS, disk use, and disk latency on the servers running MongoDB, as collected by the Automation Agent. You must run Ops Manager Automation to retrieve disk measurements.

Get Database Measurements

You must include the `granularity` parameter and either the `period` or `start` and `end` parameters.

```
GET /api/public/v1.0/groups/GROUP-ID/hosts/HOST-ID/databases/DATABASE-NAME/measurement
```

Database measurements provide statistics on database performance and storage. The Monitoring Agent collects database measurements through the dbStats command.

Get a List of Measurement Types

To retrieve the **applicable** Measurement Types without returning a large document, issue the following `GET` command with a value of `PT5M` for both the `granularity` and `period`. This returns a document with only one data point for each measurement.

```
GET /api/public/v1.0/groups/GROUP-ID/hosts/HOST-ID/measurements?granularity=PT5M&perio
```

Ops Manager filters out any measurement types that are **not** applicable. For example, if you are querying a replica set's primary, Ops Manager will not return measurements specific to replica set secondaries, such as replication lag.

Sample Entity

```
{
  "groupId" : "533c5895b910306f21033a",
  "hostId" : "1db152b29e319f270e4c34b52b01fd2e",
  "start" : "2016-08-23T17:47:44Z",
  "end" : "2016-08-23T22:22:44Z",
  "granularity" : "PT5M",
  "databaseName" : "products",
  "measurements" : [
    {
      "dataPoints" : [ {
          "timestamp" : "2016-08-23T17:47:44Z",
          "value" : 0.46137412902281843
        },
        ...
      ],
      "name" : "PROCESS_NORMALIZED_CPU_CHILDREN_KERNEL",
      "units" : "PERCENT"
    }
  ],
  "links" : [ ... ]
}
```

Entity Fields

Name	Type	Description
`groupId`	string	ID of the group that owns the host.

Name	Type	Description
`hostId`	string	ID of the host to which the measurements pertain.
`start`	string	The beginning of the period for which to retrieve measurements, specified as an ISO-8601 timestamp.
`end`	string	The end of the period for which to retrieve measurements, specified as an ISO-8601 timestamp.
`granularity`	string	An ISO-8601-formatted time period that specifies the size of the interval covered by each data point. For example, `PT5M` specifies a 5-minute granularity.
`databaseName`	string	The database to which the measurement applies. Only present for the following endpoint and its children: `../groups/GROUP-ID/hosts/HOST-ID/databases/DATABASE-NAME/measurements`
`measurements`	object array	An array of measurements and their data points.
`measurements.dataPoints`	object array	An array of objects, where each object represents a single data point. If there is no data point available for a particular moment in time (i.e., a timestamp), the `value` field is set to `null`.
`measurements.dataPoints.timestamp`	string	The timestamp of the beginning of the time interval represented by this data point.
`measurements.dataPoints.value`	float	The value of the data point.
`measurements.name`	string	The name of the measurement. For possible values, see Measurement Types below on this page.
`measurements.units`	string	How the measurement is quantified. Possible units are: • `PERCENT` • `MILLISECONDS` • `BYTES` • `GIGABYTES` • `BYTES_PER_SECOND` • `MEGABYTES_PER_SECOND` • `GIGABYTES_PER_HOUR` • `SCALAR_PER_SECOND` • `SCALAR`

Measurement Types

The `measurements` endpoint returns measurement types in the `measurements.name` field.

Host Measurements

• ASSERT_REGULAR • ASSERT_WARNING • ASSERT_MSG • ASSERT_USER	Measure the rate of asserts for a MongoDB process, as collected from the MongoDB serverStatus command's `asserts` document.
• BACKGROUND_FLUSH_AVG	Measurement found on the host's `background flush avg` chart. To view the chart, see View Metrics.
• CACHE_BYTES_READ_INTO • CACHE_BYTES_WRITTEN_FROM • CACHE_USAGE_DIRTY • CACHE_USAGE_USED • TICKETS_AVAILABLE_READS • TICKETS_AVAILABLE_WRITES	Apply to a MongoDB process's WiredTiger storage engine, as collected from the MongoDB serverStatus command's `wiredTiger.cache` and `wiredTiger.concurrentTransactions` documents.
• CONNECTIONS	Measures connections to a MongoDB process, as collected from the MongoDB serverStatus command's `connections` document.
• CURSORS_TOTAL_OPEN • CURSORS_TOTAL_TIMED_OUT	Measure the number of cursors for a MongoDB process, as collected from the MongoDB serverStatus command's `metrics.cursor` document.
• EXTRA_INFO_PAGE_FAULTS • GLOBAL_ACCESSES_NOT_IN_MEMORY • GLOBAL_PAGE_FAULT_EXCEPTIONS_THROWN	Measurements found on the host's `Record Stats` and `Page Faults` charts. To view the charts, see View Metrics.
• GLOBAL_LOCK_CURRENT_QUEUE_TOTAL • GLOBAL_LOCK_CURRENT_QUEUE_READERS • GLOBAL_LOCK_CURRENT_QUEUE_WRITERS	Measure operations waiting on locks, as collected from the MongoDB serverStatus command. Ops Manager computes these values based on the type of storage engine.
• GLOBAL_LOCK_PERCENTAGE	Applicable only to hosts running MongoDB 2.0 and earlier. Measures operations waiting on the global lock, as collected from the MongoDB serverStatus command.
• INDEX_COUNTERS_BTREE_ACCESSES • INDEX_COUNTERS_BTREE_HITS • INDEX_COUNTERS_BTREE_MISSES • INDEX_COUNTERS_BTREE_MISS_RATIO	Measurements found on the host's `btree` chart. To view the chart, see View Metrics.
• JOURNALING_COMMITS_IN_WRITE_LOCK • JOURNALING_MB • JOURNALING_WRITE_DATA_FILES_MB	Measurements found on the host's `journal - commits in write lock` chart and `journal stats` chart. To view the charts, see View Metrics.

• MEMORY_RESIDENT • MEMORY_VIRTUAL • MEMORY_MAPPED • COMPUTED_MEMORY	Measure memory for a MongoDB process, as collected from the MongoDB serverStatus command's `mem` document.
• NETWORK_BYTES_IN • NETWORK_BYTES_OUT • NETWORK_NUM_REQUESTS	Measure throughput for MongoDB process, as collected from the MongoDB serverStatus command's `network` document.
• OPLOG_SLAVE_LAG_MASTER_TIME • OPLOG_MASTER_TIME • OPLOG_MASTER_LAG_TIME_DIFF • OPLOG_RATE_GB_PER_HOUR	Measurements that apply to the MongoDB process's oplog.
• DB_STORAGE_TOTAL • DB_DATA_SIZE_TOTAL	Measurements displayed on the host's `db storage` chart. To view the chart, see View Metrics.
• OPCOUNTER_CMD • OPCOUNTER_QUERY • OPCOUNTER_UPDATE • OPCOUNTER_DELETE • OPCOUNTER_GETMORE • OPCOUNTER_INSERT	Measure the rate of database operations on a MongoDB process since the process last started, as collected from the MongoDB serverStatus command's `opcounters` document.
• OPCOUNTER_REPL_CMD • OPCOUNTER_REPL_UPDATE • OPCOUNTER_REPL_DELETE • OPCOUNTER_REPL_INSERT	Measure the rate of database operations on MongoDB secondaries, as collected from the MongoDB serverStatus command's `opcountersRepl` document.
• HOTTEST_LOCK_PERCENTAGE	Applicable only to hosts running MongoDB versions 2.2 through 2.6. Measures the amount of time hosts are write locked. For more information see `Lock %`.

Process Measurements

• PROCESS_CPU_USER • PROCESS_CPU_KERNEL • PROCESS_CPU_CHILDREN_USER • PROCESS_CPU_CHILDREN_KERNEL	The CPU usage of MongoDB. For servers with more than 1 CPU core, these values can exceed 100%. Only available if you use Ops Manager Automation.

• PROCESS_NORMALIZED_CPU_USER • PROCESS_NORMALIZED_CPU_KERNEL • PROCESS_NORMALIZED_CPU_CHILDREN_USER • PROCESS_NORMALIZED_CPU_CHILDREN_KERNEL	The CPU usage of MongoDB, scaled to a range of 0-100% by dividing by the number of CPU cores. Only available if you use Ops Manager Automation.

System Measurements

• SYSTEM_CPU_USER • SYSTEM_CPU_KERNEL • SYSTEM_CPU_NICE • SYSTEM_CPU_IOWAIT • SYSTEM_CPU_IRQ • SYSTEM_CPU_SOFTIRQ • SYSTEM_CPU_GUEST • SYSTEM_CPU_STEAL	CPU usage of processes on the host server. For servers with more than 1 CPU core, this value can exceed 100%. Only available if you use Ops Manager Automation.
• SYSTEM_NORMALIZED_CPU_USER • SYSTEM_NORMALIZED_CPU_KERNEL • SYSTEM_NORMALIZED_CPU_NICE • SYSTEM_NORMALIZED_CPU_IOWAIT • SYSTEM_NORMALIZED_CPU_IRQ • SYSTEM_NORMALIZED_CPU_SOFTIRQ • SYSTEM_NORMALIZED_CPU_GUEST • SYSTEM_NORMALIZED_CPU_STEAL	CPU usage of processes on the host server, scaled to a range of 0-100% by dividing by the number of CPU cores. Only available if you use Ops Manager Automation.

Disk Measurements

• DISK_PARTITION_IOPS_READ • DISK_PARTITION_IOPS_WRITE • DISK_PARTITION_IOPS_TOTAL	Measures throughput of I/O operations for the disk partition used for MongoDB. Only available if you use Ops Manager Automation.
• DISK_PARTITION_UTILIZATION	The percentage of time during which requests are being issued to and serviced by the partition. This includes requests from any process, not just MongoDB processes. Only available if you use Ops Manager Automation.
• DISK_PARTITION_LATENCY_READ • DISK_PARTITION_LATENCY_WRITE	Measures latency per operation type of the disk partition used by MongoDB. Only available if you use Ops Manager Automation.

• DISK_PARTITION_SPACE_FREE • DISK_PARTITION_SPACE_USED • DISK_PARTITION_SPACE_PERCENT_FREE • DISK_PARTITION_SPACE_PERCENT_USED	Measures the free disk space and used disk space on the disk partition used by MongoDB. Only available if you use Ops Manager Automation.

Database Measurements

• DATABASE_WRITE_LOCK_PERCENTAGE	Measures the amount of time the host is write locked .
• DATABASE_AVERAGE_OBJECT_SIZE • DATABASE_COLLECTION_COUNT • DATABASE_DATA_SIZE • DATABASE_FILE_SIZE • DATABASE_STORAGE_SIZE • DATABASE_INDEX_SIZE • DATABASE_INDEX_COUNT • DATABASE_EXTENT_COUNT • DATABASE_OBJECT_COUNT	Measures the database's on-disk storage space, as collected from the MongoDB dbStats command. The Monitoring Agent retrieves database measurements every 20 minutes by default but adjusts frequency when necessary to reduce the impact on database performance. You can disable the collection of database statistics through the Ops Manager interface by clicking **Settings** in the Ops Manager interface, then clicking **Group Settings**, and then setting **Collect Database Specific Statistics** to No.

Links

Relation	Description
self	Me
http://mms.mongodb.com/group	The group the host belongs to.
http://mms.mongodb.com/host	The host the measurements pertain to.
http://mms.mongodb.com/measurements/disk/partition	The disk partition the measurements pertain to. This is available for disk measurements only.
http://mms.mongodb.com/measurements/database	The database the measurement pertain to. This is available for database measurements only.

Examples

Get Host, Process, and System Measurements

Request

```
curl -i -u "username:apiKey" --digest "https://<ops-manager-host>/api/public/v1.0/grou
```

Response

```
HTTP/1.1 200 OK

{
  "end" : "2016-08-10T20:47:41Z",
  "granularity" : "PT10M",
  "groupId" : "533c5895b910306f21033a",
  "hostId" : "814e70da8167883b9939608a12a",
  "links" : [ ... ],
  "measurements" : [ {
    "dataPoints" : [ {
      "timestamp" : "2016-08-10T20:47:41Z",
      "value" : 0.0
    }, ... ],
    "name" : "ASSERT_REGULAR",
    "units" : "SCALAR_PER_SECOND"
  }, ..., {
    "dataPoints" : [ {
      "timestamp" : "2016-08-10T20:47:39Z",
      "value" : 1.3203959480720555
    }, ... ],
    "name" : "SYSTEM_NORMALIZED_CPU_STEAL",
    "units" : "PERCENT"
  } ],
  "start" : "2016-08-08T20:57:38Z"
}
```

Get Disk Partition Measurements

Request

```
curl -i -u "username:apiKey" --digest "https://<ops-manager-host>/api/public/v1.0/grou
```

Response

```
HTTP/1.1 200 OK

{
  "end" : "2016-08-10T03:40:38Z",
  "granularity" : "PT5M",
  "groupId" : "533c5895b910306f21033a",
  "hostId" : "814e70da8167883b9939608a12a",
  "links" : [ ... ],
  "measurements" : [ {
    "dataPoints" : [ {
      "timestamp" : "2016-08-10T03:40:38Z",
      "value" : 0.0
    }, ... ],
    "name" : "DISK_PARTITION_IOPS_READ",
    "units" : "SCALAR_PER_SECOND"
  }, ..., {
    "dataPoints" : [ {
      "timestamp" : "2016-08-10T03:40:38Z",
      "value" : 3.2846554854156516
    }, ... ],
    "name" : "DISK_PARTITION_SPACE_PERCENT_USED",
    "units" : "PERCENT"
  } ],
  "partitionName" : "xvdf",
  "start" : "2016-08-09T03:50:21Z"
}
```

Get Database Measurements

Request

```
curl -i -u "username:apiKey" --digest "https://<ops-manager-host>/api/public/v1.0/grou
```

Response

```
HTTP/1.1 200 OK

{
  "databaseName" : "markets",
  "end" : "2016-08-11T21:08:40Z",
  "granularity" : "PT5M",
  "groupId" : "533c5895b910306f21033a",
  "hostId" : "fd5b59188dc13ad142493",
  "measurements" : [ {
    "dataPoints" : [ ],
    "name" : "DATABASE_WRITE_LOCK_PERCENTAGE",
    "units" : "PERCENT"
  },
  {
    "dataPoints" : [ {
      "timestamp" : "2016-08-11T21:08:40Z",
      "value" : 51.780589415213704
    } ... ],
    "name" : "DATABASE_AVERAGE_OBJECT_SIZE",
    "units" : "BYTES"
  },
  ...,
  {
    "dataPoints" : [ {
      "timestamp" : "2016-08-11T21:08:40Z",
      "value" : 32405.0
    } ],
    "name" : "DATABASE_OBJECT_COUNT",
    "units" : "SCALAR"
  } ],
  "start" : "2016-08-11T21:08:40Z",
  "links" : [ ... ]
}
```