Foglight for MongoDB

Cartridge Guide

The information in this document is provided in connection with Quest Software products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Quest Software products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, QUEST SOFTWARE ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL QUEST SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF QUEST SOFTWARE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Quest Software makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Quest Software does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

Quest Software Inc.

Attn: LEGAL Dept

4 Polaris Way

Aliso Viejo, CA 92656

Refer to our Web site (https://www.quest.com) for regional and international office information.

Patents

Quest Software is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at https://www.quest.com/legal .

Trademarks

Quest, the Quest logo, and Join the Innovation are trademarks and registered trademarks of Quest Software Inc. For a complete list of Quest marks, visit https://www.quest.com/legal/trademark-information.aspx. All other trademarks and registered trademarks are property of their respective owners.

Table of Contents

Introduction

Description

MongoDB is a cross-platform document-oriented database. MongoDB falls into the class of a NoSQL database and eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas (MongoDB calls the format BSON). Instead of taking a business subject and breaking it up into multiple relational structures, MongoDB can store the business subject in the minimal number of documents. For example, instead of storing title and author information in two distinct relational structures, title, author, and other title-related information can all be stored in a single document called Book. This has the benefit of making the integration of data in certain types of applications easier and faster. Released under the “Server Side Public License”, MongoDB Server is open-source software.

First developed by the software company MongoDB Inc. in October 2007 as a component of a planned platform as a service product, the company shifted to an open-source development model in 2009, with MongoDB offering commercial support and other services. Since then, MongoDB has been adopted as backend software by a number of major websites and services, including Craigslist, eBay, and Foursquare among others. As of July 2015, MongoDB is the fourth most popular type of database management system, and the most popular for document stores.

Business Challenge

MongoDB is built to be highly scalable and deployments regularly scale over a number of servers and across network segments. MongoDB scales horizontally using sharding. The user chooses a shard key, which determines how the data in a collection will be distributed. The data is split into ranges (based on the shard key) and distributed across multiple shards. (A shard is a master with one or more slaves.) MongoDB can run over multiple servers, balancing the load and/or duplicating data to keep the system up and running in case of hardware failure. Mongo’s architecture makes it highly scalable and capable of running in environments requiring 24/7 availability. The downside to this complex architecture is that many elements need to be monitored and tuned to ensure optimum performance and availability.

Key Features

Foglight for MongoDB© is designed to provide enterprises with a powerful tool that can act standalone or as part of broader cross-platform database monitoring requirements. To support enterprise performance Foglight for MongoDB© will support the monitoring of key functions including the MongoDB Server, Queries, Indexing, Replication, Load Balancing and File Storage.

As with all Foglight solutions, Foglight for MongoDB© will run as a native Foglight process leveraging the Foglight Agent Manager (FglAM) for communications with the agent. Foglight for MongoDB can be run locally (Agent on the Host) or remotely (Agent on FMS or Proxy Server). The solution will leverage the Foglight Rules Engine providing the ability to evaluate and alert on potential issues before they significantly affect performance and availability.

Foglight for MongoDB Requirements

MongoDB Version Requirements

Foglight for MongoDB is compatible with **MongoDB 2.4+**. However, some data may not be available in earlier versions. The following list itemizes which features become available at which versions.

v2.6.0:

User and role data available.

v3.0.0:

Replica set configuration available.

Log cache sample available.

v3.3.12:

Autosplit enabled metric available for sharded clusters.

v3.4.4:

Profiled operation command is available. Examples include "find", "dbStats", and "getMore". The command is used as a grouping field when aggregating operations for Profiled Operation Historical Aggregates. Prior to this version, grouping is by namespace and operation only.

v4.2.0:

Profiled operation query hash is available. The hash is used as a grouping field when aggregating operations for Profiled Operation Historical Aggregates. Prior to this version, the command takes the place of query hash as a group-by field.

Foglight Version Requirements

Foglight for MongoDB can be installed on **FMS 7.1.0.10+** and agents require **FglAM 7.1.0.10+**.

Installing and Configuring Agents

Installation of Foglight for MongoDB is covered in the following sections and should be performed in order:

- MongoDB Server Pre-Configuration
- Cartridge Installation
- Creating and Configuring Agents

MongoDB Server Pre-Configuration

In order to allow full monitoring of the MongoDB Server, the agent will require a user with sufficient privilege to execute system queries. Additional steps are required to allow a TLS/SSL connection with the agent if the database server is configured to support or require this. In order to collect profiled operation data, profiling must be enabled on each MongoDB instance on a per-database basis.

MongoDB Agent User Permissions

The MongoDB agent requires database user credentials with certain minimum privileges in order to be able to fully monitor the server and cluster. All user authorizations must be for the ‘admin’ database. The roles needed for the user will vary depending on the MongoDB version. Note that when monitoring a sharded cluster, a database user will need to be created separately on each component replica set, i.e. on the config server replica set as well as on each shard. The code examples for creating a database user assume the default authentication method. For non-default authentication methods (such as x.509 and LDAP, for which the user is created on the ‘$external’ database) these examples may need to be modified.

For MongoDB 3.0 and later the following roles on the admin database are required:

```
'clusterMonitor', 'readAnyDatabase'
```

User creation example for MongoDB 3.0 and later:

```
db.getSiblingDB('admin').createUser({
    user: 'foglightAgent',
    pwd: '<xxxx>',
    roles: ['clusterMonitor', 'readAnyDatabase']
})
```

For versions prior to MongoDB 3.0 the following roles are required:

```
'clusterAdmin', 'dbAdminAnyDatabase', 'readAnyDatabase'
```

User creation example prior to MongoDB 3.0:

```
db.getSiblingDB('admin').addUser({
    user: 'foglightAgent',
    pwd: '<xxxx>',
    roles: ['clusterAdmin', 'dbAdminAnyDatabase', 'readAnyDatabase']
})
```

Additional privileges are required for monitoring users and roles, which is available starting with MongoDB v2.6. Specifically, the "`viewUser`" and "`viewRole`" privilege actions are required on any databases for which user and role data is to be collected. Following is an example script to create a new role enabling monitoring users and roles on all databases and grant it to the Foglight monitoring user:

```
db.getSiblingDB('admin').createRole({
    role: 'viewAllUsersAndRoles',
    privileges: [{ resource: { db: '', collection: '' }, actions: ['viewUser', 'viewRole'] }],
    roles: []
})
db.getSiblingDB('admin').grantRolesToUser('foglightAgent', ['viewAllUsersAndRoles'])
```

Finally, ensure that all firewalls and network configurations allow the machine running the FglAM to access the host of each monitored server at its configured port.

Configuring an SSL Connection

The below instructions cover common steps used to configure a TLS/SSL connection from the MongoDB Agent client. For full information on secure connections and server-side configuration, refer to the TLS/SSL Configuration for Clients section of the MongoDB documentation for your database version.

A full treatment of TLS/SSL keys, certificates, and certificate authorities (CA) is beyond the scope of this document. The following instructions assume familiarity with TLS/SSL concepts and tools. Client and certificate authority certificates must be available prior to proceeding.

In order to use SSL, your MongoDB server must include SSL support and allow SSL connections. There are various configurations options for client connections. Refer to the MongoDB documentation and verify that the current MongoDB server configuration parameters support the desired authentication.

The Foglight agent, in its capacity as a database client, requires access to a private key, its signed certificate, and the signing CA's certificate. The client key and certificate must be imported into a keystore, and the CA certificate must be imported into a separate truststore.

One example method for generating a JKS keystore for use with Foglight utilizes openssl and keytool. Set the key and certificate filenames, alias name, and keystore password as appropriate.

```
openssl pkcs12 \
        -export \
        -in ${CERT_NAME}.crt \
        -inkey ${CERT_NAME}.key \
        -name $CERT_NAME \
        -out temp-keystore.p12 \
        -passout pass:${KEYPASS}

keytool -importkeystore \
        -srckeystore temp-keystore.p12 \
        -srcstoretype PKCS12 \
        -srcstorepass $KEYPASS \
        -destkeystore keystore \
        -deststoretype JKS \
        -deststorepass $KEYPASS
```

Regardless of how the keystore is constructed, it must list the client certificate as a 'PrivateKeyEntry', indicating that it also contains the private key, not just the signed certificate.

Separately, import the CA certificate into a truststore:

```
keytool -importcert \
        -keystore truststore \
        -alias $CA_NAME \
        -file ${CA_NAME}.crt \
        -keypass $TRUSTPASS \
        -storepass $TRUSTPASS \
        -storetype JKS \
        -noprompt
```

Next, edit the baseline.jvmargs.config file in the <FglAM-install-root>/state/default/config directory and add the following parameters with file paths and passwords appropriate for your system.

```
vmparameter.0 = "-Djavax.net.ssl.keyStore=/path/to/keystore";
vmparameter.1 = "-Djavax.net.ssl.keyStorePassword=changeit";
vmparameter.2 = "-Djavax.net.ssl.trustStore=/path/to/truststore";
vmparameter.3 = "-Djavax.net.ssl.trustStorePassword=changeit";
```

Escape any quotes with a backslash (‘\’). On an Agent Manager installed on Windows, use forward slashes in the file paths, like so:

```
vmparameter.0 = "-Djavax.net.ssl.keyStore=\"C:/path/to/keystore\"";
```

Then, restart the FglAM and continue with the agent configuration, setting the “Use TLS/SSL?” option in the Agent Properties to true. If the client certificate is not configured specifically for the FglAM host, you can also set the “Allow Invalid Cert Hostname?” option to true to allow the certificate to be used anyway.

Enabling Profiling

Operation profiling is disabled by default on MongoDB servers. If profiling information is desired for monitoring purposes, profiling must be enabled in MongoDB. Profiling is enabled on an instance-by-instance and database-by-database basis. So if, for example, profiling for a specific database is desired across a three instance replica set, profiling can be enabled on the primary only or on all three instances, as desired. Profiling on MongoDB does impact database performance, and should be used judiciously in production environments. For more information see Database Profiling Overhead.

To enable profiling, first connect to the MongoDB instance with the mongo shell. The database user must be assigned a role with the `enableProfiler` privilege action on the respective database. An example of a built-in role that permits profiler enabling on non-system databases is `dbAdmin`.

Switch to the database for which profiling should be enabled. Replace “`<db-name>`” below with the name of the database:

```
use <db-name>
```

To check the current profiling status, execute:

```
db.getProfilingStatus()
```

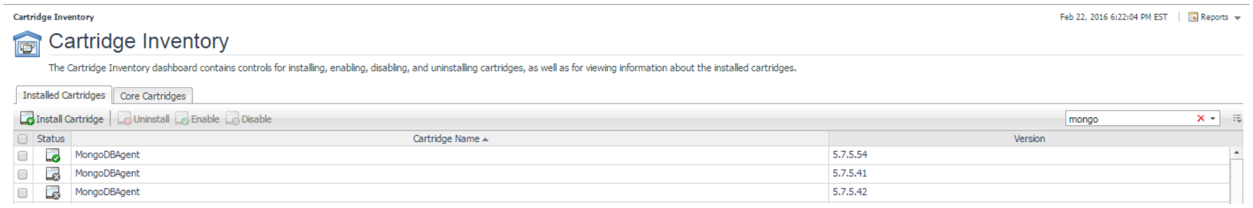
Modify the profiling level by executing the `setProfilingLevel` command. For example, the following command will enable profiling for all operations taking longer than 250 milliseconds:

```
db.setProfilingLevel(1, { slowms: 250 })
```

This is only one example of using the `setProfilingLevel` command and should not be taken as a recommendation for your database. For full details and options for using this command, see setProfilingLevel.

Cartridge Installation

1. Open Foglight Management Console.
2. From the navigation pane, select: **Dashboards** > **Administration** > **Cartridges** > **Cartridge Inventory**. The Cartridge Inventory screen appears. For more information on agents, see the *Foglight User Guide.*
3. Load the ***MongoDBAgent-xxxx.car*** file by browsing to the location where the .car file exists and then clicking on "Install Cartridge". Leave the "Enable on Install" check box checked.
4. Once the installation is completed on the Foglight Management Server, the MongoDBAgent Cartridge will appear in this list below as an installed cartridge.



Cartridge Inventory

The Cartridge Inventory dashboard contains controls for installing, enabling, disabling, and uninstalling cartridges, as well as for viewing information about the installed cartridges.

□	Status	Cartridge Name ▲	Version
□	✓	MongoDBAgent	5.7.5.54
□	✗	MongoDBAgent	5.7.5.41
□	✗	MongoDBAgent	5.7.5.42

Creating and Configuring Agents

Agents can be created in one of two ways:

- Using the Agent Installer Wizard
- Using the Agent Status Dashboard

The Agent Installer Wizard simplifies the agent creation and configuration process and can be accessed from the Databases dashboard. For advanced configuration or modification of agent properties post-creation, use the Agent Status dashboard.

Using the Agent Installer Wizard

Foglight for MongoDB provides a graphic, intuitive method for creating and configuring agents, which can be used instead of Foglight's default method for creating agents and editing their properties using the Agent Status dashboard. Foglight for MongoDB allows running a wizard that provides a common entry point for adding database instances and then configuring these instances for monitoring.

To run the instance installation wizard:

1. On the navigation panel, click Homes > Databases.
2. Click the MongoDB box in the Databases View, and then click Monitor.
3. The Agent Installer Wizard dialog box appears.
4. The first card - Agent Deployment – has two fields:
    a. Agent Name – Provide a name for the agent that will be created. This is not canonical and should be representative of the database instance that this agent will monitor.
    b. Agent Manager - Choose the agent manager on which the agent should run. Considerations for this may include physical or virtual locality to the monitored instance, allocated resources, or grouping with other agents of the same type or monitored environment. If the agent package has not been deployed to this Agent Manager yet, it will be installed when the first agent of this type is created.
5. The second card – Agent Properties – requires a basic set of parameters for connecting to and monitoring the database instance. A full explanation of these properties is available in the Agent Properties section of this document.
6. The third card – Agent Summary – displays a review of the configuration that will be created and an option allowing the agent to be activated after creation. If the configuration looks good, click the Finish button to start the process.
7. When the process completes, a results screen will appear showing the results of agent creation. If the agent was not created, follow the instructions on the results screen. If successful, the database instance should appear in the Databases table within a few minutes.

Note: If the agent was created successfully but data is not appearing, go to the Dashboards > Administration > Agents > Agent Status page and click the icon in the Log File column for the agent you created. In most cases, the reason for the failure will be obvious. You can also refer to the *Foglight for MongoDB Installation and Troubleshooting* document for common errors and solutions. If the solution requires reconfiguring the agent properties, follow steps 3-7 of the Using the Agent Status Dashboard section.

Using the Agent Status Dashboard

The Agent Status page can be used to create new agents and configure and manage existing agents. To access the page from the navigation pane, select: Dashboards > Administration > Agents > Agent Status.

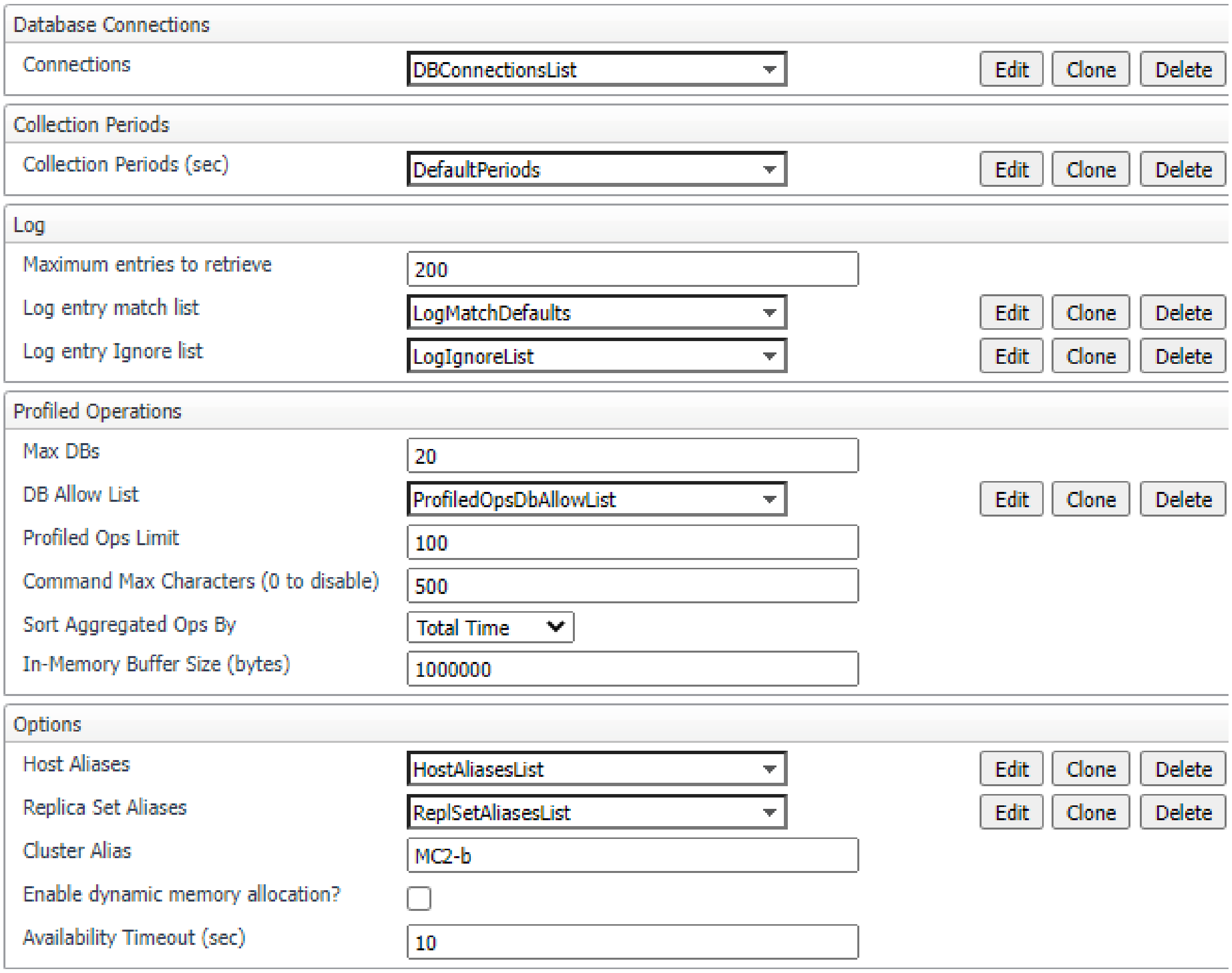
Use the following steps to create a new agent instance:

1. If the MongoDB agent package has never been deployed to the FglAM that will be used to host the agent, this must be done before an agent has been created. You can use the Deploy Agent Package button on the Agent Status or Agent Managers page to perform this.
2. Click the Create Agent button and follow the instructions for the cards:
    - **a. Host Selector** - Choose the Agent Manager on which the agent should run. Considerations for this may include physical or virtual locality to the monitored instance, allocated resources, or grouping with other agents of the same type or monitored environment.
    - **b. Agent Type and Instance Name** – Select the MongoDBAgent type. Then, select the Specify Name radio button and provide a name for the agent that will be created. This is not canonical and should be representative of the database instance that this agent will monitor.
    - c. **Summary** – Click Finish.
3. Once the agent has been created, click the checkbox next to the MongoDB agent.
4. Click the **Edit Properties** button.
5. Select **Modify the default properties for this agent**.
6. Edit the agent properties for the MongoDB agent instance:
    - Database Connections
    - Log
    - Options
7. Click the **Activate** button.

To modify the properties for an existing agent, skip to step 3 and Deactivate, then Reactivate the agents after changing the configuration.

Agent Properties

This is a full list and explanation of the configurable properties of the Foglight for MongoDB agent. The Agent Installer Wizard provides access to the essential subset of available properties. To modify other properties or modify the agent configuration after creation, use the Agent Status dashboard.



Database Connections	
Connections	DBConnectionsList
Collection Periods	
Collection Periods (sec)	DefaultPeriods
Log	
Maximum entries to retrieve	200
Log entry match list	LogMatchDefaults
Log entry Ignore list	LogIgnoreList
Profiled Operations	
Max DBs	20
DB Allow List	ProfiledOpsDbAllowList
Profiled Ops Limit	100
Command Max Characters (0 to disable)	500
Sort Aggregated Ops By	Total Time
In-Memory Buffer Size (bytes)	1000000
Options	
Host Aliases	HostAliasesList
Replica Set Aliases	ReplSetAliasesList
Cluster Alias	MC2-b
Enable dynamic memory allocation?	☐
Availability Timeout (sec)	10

Database Connections

The Connections property list contains the connection details necessary to access the database instances as well as host and display aliases, and collection intervals. Add one row for each mongos query server, config server, and mongod data server in the sharded cluster or replica set to be monitored. Do not add a row for arbiters. In each row, enter the host and port of the instance, the username and password of the database user, check whether SSL is required for access, and specify the authentication method. See below for host aliases - these can be configured as needed. Display aliases are solely for human readability and convenience, for example to shorten long alphanumeric host addresses to descriptive, quickly recognizable names such as "PROD_BillingApp_QueryServer2". Leave "Enable Monitoring" checked unless the server should be ignored for some reason.

Collection Periods

The collection periods are intervals between collections in seconds. They are used to set the sampling frequencies. A given collection can be disabled by setting its interval to zero. The defaults are set based on the type of data being collected. The following are the configurable collection periods: Connection Status, Server, Database, Collection, Top, Shard, Replication, Profiled Ops, Profiled Ops Buffer, Role, Parameters, Log.

Log

- **Maximum entries to retrieve** – The maximum number of log entries to retrieve per instance, per collection period. Set to zero to disable the log collection for all MongoDB instances monitored by the agent. To disable the log collection on an individual instance, set the Log period to zero for just that instance in the Database Connections list.
- **Log entry match list** – A list of rules specifying conditions for when log entries should cause a Foglight alert to fire. Consists of a severity level found in the log entry, a regular expression text to search for in the log entry message text and the Foglight severity at which to fire an alarm if a match is found. Any entry with a log severity more extreme than the one provided in the list will naturally also trigger an alarm. Note that the log entry match list works by submitting data through the agent that will cause the rule "MongoDB Log Patterns" to trigger. Hence if this rule is disabled, no alerts will fire on log entries even if the match list has been set up with matching conditions.
- **Log entry ignore list** – The ignore list looks much like the match list except that it prevents triggering of the rule "MongoDB Log Patterns". If a collected log entry matches a rule in the match list, it will only fire if it does not also match a rule in the ignore list.

Profiled Operations

- **Max DBs To Collect** – The maximum number of databases from which to collect profiled operations. Set to zero to disable profiled operation collection. The maximum allowed value is 100.
- **DB Allow List** – An optional list of database names to either allow or disallow. If any entry is disallowed, then data on all non-disallowed databases will be collected, otherwise only data for the specifically allowed databases will be collected. Providing an empty list allows all databases.
- **Profiled Ops Limit** – The maximum number of profiled operations to collect in each collection period.
- **Command Max Characters** – The maximum character length of the profiled operation command text to store. Set to zero to disable command text collection.
- **Sort Aggregated Ops By** – The numerical property by which to select the top profiled operations. Available metrics are: Average Time, Executions, Max Time, and Total Time.
- **In-Memory Buffer Size** – The size of the in-memory buffer used to store the profiled operations buffer data (individual operations). This buffer resides on the Foglight Agent Manager and contributes to its memory usage. Set to zero to disable the profiled operations buffer collection. The maximum allowed value is 10 MB.

Options

- **Host Aliases** – A list of hostnames or IP addresses mapped to aliases. Can be used to standardize discovered hostnames from internal representations (e.g., in a cloud environment) to their external address. Enables, for example, the MongoDB cartridge to link directly to the Hosts dashboard by ensuring host data is submitted uniformly across the monitoring environment.

- **Replica Set Aliases** – A mapping from actual replica set names to aliases. This alias map is required to prevent collisions in collected data between different replica sets with the same name. Ideally, each replica set in an environment should be set up with a unique name.
- **Cluster Alias** – An arbitrary human-friendly name to display in place of the UUID cluster ID. Only used for sharded clusters.
- **Enable Dynamic Memory Allocation?** – Select yes to enable dynamic memory allocation. In this case memory may be added to the environment as the number of monitored MongoDB collections grows. If set to false, memory allocation is performed by the agent manager on a per-agent basis only.
- **Availability Timeout (sec)** – The time to wait in seconds before a connection to a MongoDB server is considered to have failed.

Roles

Two roles, MongoDB User and MongoDB Administrator, are installed with the cartridge. Viewing MongoDB dashboards requires that a user be assigned one of these or have the core Administrator role. Currently, there are no added privileges for the MongoDB Administrator role, but future cartridge versions which allow user interaction with the MongoDB Server will require this role.

Upgrading the Agent

1. Go to Dashboards > Administration > Cartridges > Cartridge Inventory and click the Install Cartridge button.
2. Locate the .car file on your system and install it with auto-enable selected. If you get a message that a bundled cartridge is of an older version than the one currently enabled on your FMS and will not be enabled, ignore it and continue.
3. Once the cartridge is installed and enabled, go to Dashboards > Administration > Agents > Agent Managers. Agent Managers that can be upgraded with newer agent packages will show “yes” in the Upgradable | Agents column. Select all Agent Managers you wish to upgrade and click the Upgrade button.

Note: If an Agent Manager is not upgradable, check that the Agent Manager version is compatible with the newer agent version. If it is not, the Agent Manager will need to be upgraded first.

Removing Monitored Databases

1. Go to the Databases dashboard.
2. Select the databases you wish to remove.
3. Click the Settings button, then click ok.

Note: Doing this will remove the monitoring agents as well as the historical data already collected. If you wish to delete only the agents, you can do that on the Administration > Agents > Agent Status page. Because the Databases dashboard only shows databases which are being actively monitored, you will only be able to view historical data for databases whose monitoring agents have been deleted by going directly to the MongoDB dashboards.

Administration

Opening the Databases Administration Dashboard

You can edit agent settings for one or more MongoDB instances on the Databases > Administration dashboard.

> **NOTE:** If you attempt to select instances of more than one type of database, such as a MongoDB database and an Oracle database, an error message is displayed.

To open the Databases Administration dashboard:

1. In the navigation panel, under **Homes**, click **Databases > MongoDB**.
2. Select the check boxes beside one or more MongoDB instances.
3. Click **Settings** and then click **Administration**. The Administration dashboard opens, containing settings for all the selected agents. Settings are broken down into categories, which are organized under a MongoDB tree.

Reviewing the Administration Settings

The Databases Administration dashboard allows settings options for collecting, storing, and displaying data, which apply to all the currently selected agents. Click a category of settings on the left (for example: Connection Details) to open a view containing related settings on the right.

To view the full list of selected agents, click the **Selected Agents** button at the upper right corner of the screen. To change the list of agents to which the metrics will apply, exit the Databases Administration dashboard, select the requested agents and re-open the view.

Customizing Alarms for Foglight for MongoDB Rules

Many Foglight for MongoDB multiple-severity rules trigger alarms. To improve your monitoring experience, you can customize when alarms are triggered and whether they are reported. You can also set up email notifications.

Introducing the Alarms View

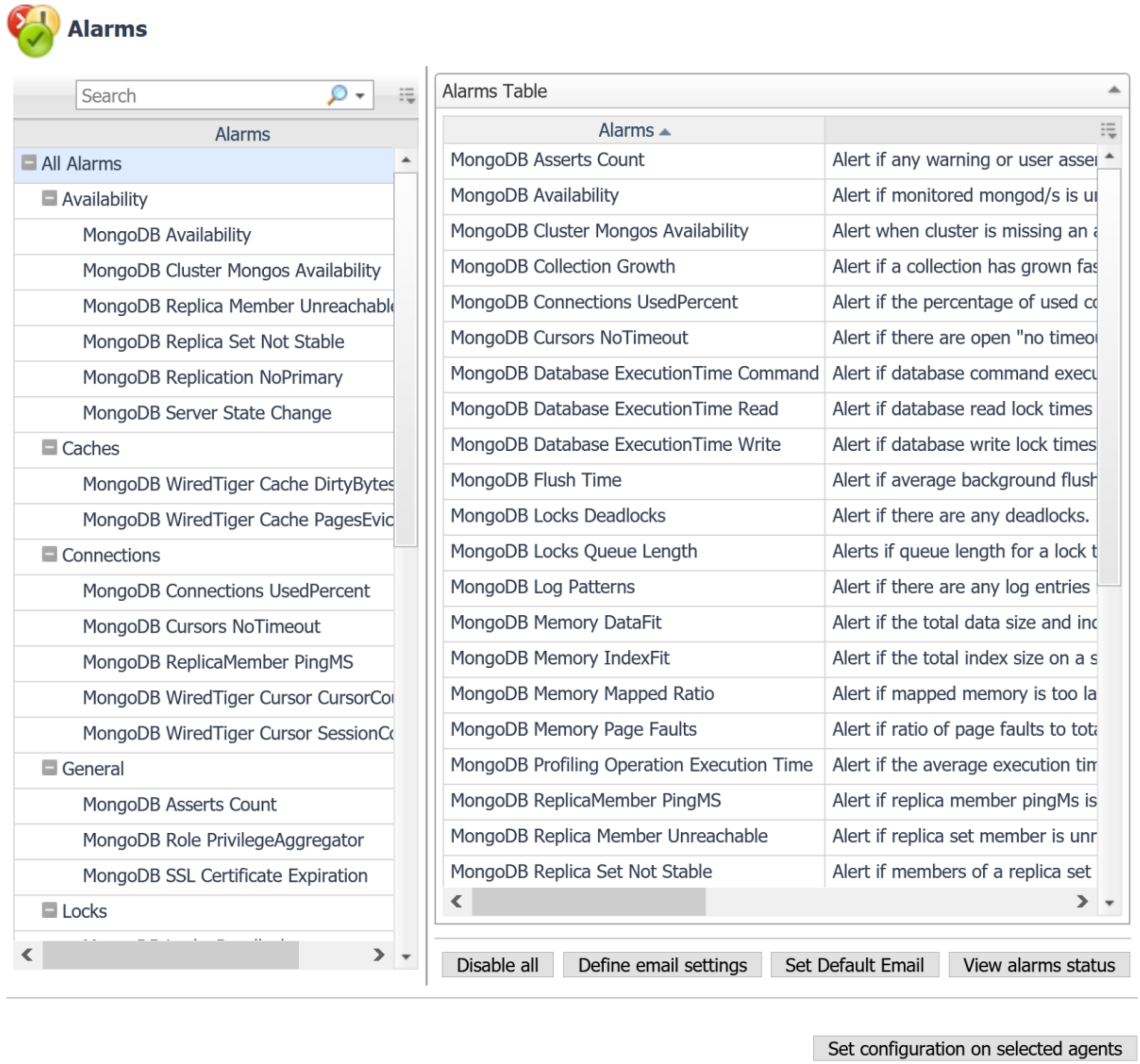
The Alarms view enables you to modify global settings and agent-specific settings for alarms.

To open the Alarms view:

1. Open the Administration dashboard as described in Opening the Databases Administration Dashboard.
2. Select the agents you wish to modify and do one of the following steps:
    a. Select the Settings button and open the Administration dashboard, then click Alarms.
    b. Select the 'Configure Alarm' button.
3. From the Alarms view, you can complete the following tasks:
    a. Modifying Alarm Settings
    b. Reviewing Rule Definitions
    c. Cloning Agent Settings

Modifying Alarm Settings

You can customize how the alarms generated by the default rules are triggered and displayed in the Alarm view. Changes to alarm settings will apply to all selected agents, though thresholds can be customized by individual agent.



Alarms

- All Alarms
  - Availability
    - MongoDB Availability
    - MongoDB Cluster Mongos Availability
    - MongoDB Replica Member Unreachabl
    - MongoDB Replica Set Not Stable
    - MongoDB Replication NoPrimary
    - MongoDB Server State Change
  - Caches
    - MongoDB WiredTiger Cache DirtyBytes
    - MongoDB WiredTiger Cache PagesEvic
  - Connections
    - MongoDB Connections UsedPercent
    - MongoDB Cursors NoTimeout
    - MongoDB ReplicaMember PingMS
    - MongoDB WiredTiger Cursor CursorCo
    - MongoDB WiredTiger Cursor SessionC
  - General
    - MongoDB Asserts Count
    - MongoDB Role PrivilegeAggregator
    - MongoDB SSL Certificate Expiration
  - Locks

Alarms ▲	
MongoDB Asserts Count	Alert if any warning or user asser
MongoDB Availability	Alert if monitored mongod/s is u
MongoDB Cluster Mongos Availability	Alert when cluster is missing an
MongoDB Collection Growth	Alert if a collection has grown fas
MongoDB Connections UsedPercent	Alert if the percentage of used c
MongoDB Cursors NoTimeout	Alert if there are open "no timeo
MongoDB Database ExecutionTime Command	Alert if database command execu
MongoDB Database ExecutionTime Read	Alert if database read lock times
MongoDB Database ExecutionTime Write	Alert if database write lock times
MongoDB Flush Time	Alert if average background flush
MongoDB Locks Deadlocks	Alert if there are any deadlocks.
MongoDB Locks Queue Length	Alerts if queue length for a lock t
MongoDB Log Patterns	Alert if there are any log entries
MongoDB Memory DataFit	Alert if the total data size and inc
MongoDB Memory IndexFit	Alert if the total index size on a s
MongoDB Memory Mapped Ratio	Alert if mapped memory is too la
MongoDB Memory Page Faults	Alert if ratio of page faults to tot
MongoDB Profiling Operation Execution Time	Alert if the average execution tir
MongoDB ReplicaMember PingMS	Alert if replica member pingMs is
MongoDB Replica Member Unreachable	Alert if replica set member is unr
MongoDB Replica Set Not Stable	Alert if members of a replica set

The Alarms list controls the contents displayed to the right and the tasks that are available.

- **All Alarms –** Displays all rules with configured alarms and indicates whether alarms are enabled. In this view, you can enable or disable alarms for all the rules at once. You can also set email notifications and define mail server settings.
- **Category of rules –** Displays a set of related rules with configured alarms. In this view, you can enable or disable alarms and also set email notifications for the category of rules.
- **Rule name –** Displays the alarm status for the selected rule. If the rule has multiple severity levels, displays the threshold configured for each severity level. In this view, you can enable or disable the alarm, edit the alarm text, and edit severity levels and their thresholds. You can also set email notifications for the alarm.

You can complete the following tasks:

- Enabling or disabling alarms for selected agents
- Modifying alarm threshold values
- Editing the text of the alarm message

Your changes are saved separately and applied over the default rules. This protects you from software upgrades that may change the underlying default rules.

Enabling or disabling alarms for selected agents

You can override the global alarm sensitivity level setting for the selected agents. You can enable or disable alarms for all rules, a category of rules, or an individual rule.

To see descriptions of the rules, follow the steps described in Reviewing Rule Definitions.

To enable or disable alarms:

1. Navigate to the Alarms view.
2. Decide on the scope for the change: all alarms, a category of rules, or a selected rule.
3. Complete the steps for the selected scope:

Scope	Procedure
All alarms	Click **All Alarms**. In the Alarms Settings tab, click either **Enable all** or **Disable all**.
Category of rules	Click a category. Click either **Enable all** or **Disable all**.
Selected rule	Click the rule. In the Alarms Settings tab, click the link that displays the alarm status. Select **Enabled** or **Disabled** from the list and click **Set**.

Modifying alarm threshold values

You can and should modify the thresholds associated with alarms to better suit your environment. If you find that alarms are firing for conditions that you consider to be acceptable, you can change the threshold values that trigger the alarm. You can also enable or disable severity levels to better suit your environment.

When a rule has severity levels, a Threshold section appears in the Alarm Settings tab showing the severity levels and bounds by agent. The threshold values correspond to the lower bounds shown in this table. Many rules do not have severity levels and thresholds.

When editing thresholds, ensure that the new values make sense in context with the other threshold values. For most metrics, threshold values are set so that Warning < Critical < Fatal. However, in metrics where normal performance has a higher value, such as DBSS - Buffer Cache Hit Rate, the threshold values are reversed: Warning > Critical > Fatal.

To change severity levels and thresholds:

1. Navigate to the Alarms view.
2. Click the multiple-severity rule that you want to edit.
3. Click the **Alarms Settings tab.**
4. In the Threshold section, review the defined severity levels and existing threshold bounds for all target agents.
5. Modify the severity levels for one or more agents by following one of the following procedures:

Task	Procedure
Edit severity levels and set threshold values for all agents.	Click **Enhance alarm**. Select the check boxes for the severity levels you want enabled and set the threshold values. Click **Set**.
Change the threshold values for one agent.	Click **Edit** beside the agent name. Set the new threshold values and click **Set**.
Copy the changes made to one agent's threshold values to all other agents.	Click **Edit** beside the agent name that has the values you want to copy. Select **Set for all agents in table** and click **Set**.

Editing the text of the alarm message

For individual rules, you can change the message displayed when an alarm fires. You cannot add or remove the variables used in the message. This is a global setting that affects all agents.

To change the alarm message:

1. In the Alarms view, click the **Settings** tab.
2. Select a rule.
3. Click the **Alarm Settings** tab.
4. Click **Enhance alarm**. A Customize <rule> dialog box opens.
5. In the Message box, edit the message text. To restore the default message, click **Reset message**.
6. Click **Set**.

Reviewing Rule Definitions

If you want to review the conditions of a rule, open the rule in the Rule Management dashboard.

> **IMPORTANT:** Avoid editing rules in the Rule Management dashboard unless you are creating your own rules or copies. These rules may be modified during regular software updates and your edits will be lost.

You can create user-defined rules from the Rule Management dashboard. If you want to modify a rule, we recommend copying the rule and creating a user-defined rule. User-defined rules need to be managed from the Rule Management dashboard; these rules are not displayed in the Alarms view of the Databases Administration dashboard. For help creating rules, open the online help from the Rule Management dashboard.

To open the Rule Management dashboard:

1. On the navigation panel, under **Homes**, click **Administration**.
2. In the Administration dashboard, click **Rules**.
3. Type **MongoDB** in the Search field to see the list of predefined rules for MongoDB databases. The MongoDB rules are displayed. From here, you can review threshold values, alarm counts, and descriptions.
4. To see the full rule definition, click a rule and then click **View and Edit**.
5. In the Rule Detail dialog box, click **Rule Editor**.
6. When you are done with your review, click Rule Management in the bread crumbs to return to the dialog box.
7. Click **Cancel** to avoid changing the rule unintentionally.

Cloning Agent Settings

You may want an agent to have the same settings as another agent. For example, if you add new agents, you may want them to use the same settings as an existing agent. In this case, you can clone the settings from one agent to other agents. This process does not link the agents; in the future if you update the source agent, you also need to update the target agents.

This procedure walks you through selecting the source agent from the Databases dashboard. However, you can also open the Administration dashboard with multiple agents selected. In this case, you select the source agent in Clone Alarm-related Settings to Other Agents dialog box.

To clone alarm-related settings:

1. On the Databases dashboard, select the check box for the agent with the settings you want to clone.
2. Click Settings and then Administration.
3. In the Administration dashboard, click Alarms.
4. Click Set configuration on selected agents. The Clone rule settings across agents dialog box opens.
5. In the Select the source agent drop-down list, you should see the agent you selected.
6. In the Select the target agents table, select the check boxes for agents that should inherit settings from the source agent.
7. Click Apply.
8. When prompted for confirmation, click Yes.

Configuring Email Notifications

We recommend that you set email notifications for the alarms you are most interested in tracking closely. For example, you may want to be notified by email of any Critical or Fatal situation. Or you may want to be informed whenever a key metric is no longer operating within acceptable boundaries.

You can set up email notifications that are generated when an alarm fires and/or on a defined schedule, as described in the following topics:

- Configuring an email server
- Defining Default Email settings
- Enabling or disabling email notifications
- Defining email notifications, recipients, and messages
- Defining variables to contain email recipients

Configuring an email server

You need to define the global mail server variables (connection details) to be used for sending email notifications.

The setting of the email should be configured in Foglight Administration > Email configuration.

Defining Default Email settings

You can define a default email address to be used by every new agent created in the future, by selecting the Default email button when configuring email notification.

The Email addresses entered are applied to all monitored agents not only for the agents that were selected to enter the Alarm administration.

Enabling or disabling email notifications

You can enable or disable email notifications for all alarms, a category of alarms, or a selected rule. Email notifications are sent only if all the following conditions are met:

- The alarm email notification setting is enabled for the affected rule.
- The alarm is triggered by changes in the monitored environment.
- Alarm notification is enabled at the triggered severity level. See Defining email notifications, recipients, and messages.

To enable or disable email notifications:

1. In the Alarms view, click the Settings tab.
2. Decide on the scope for the change: all alarms, a category of rules, or a selected rule.
3. Complete the steps for the selected scope:
   - All alarms - Click All Alarms. Click the Define Email Settings button. Select either Enabled or Disabled from the Alarms notification status list. Click Set.
   - Category of rules - Click a category. Click the Define Email Settings button. Select either Enabled or Disabled from the Alarms notification status list. Click Set.

- Selected rule - Click a rule. In the Alarms Settings tab, click the Define Email Settings tab. Click the link that displays the alarm notification status. Select Enabled or Disabled and click Set.

Defining email notifications, recipients, and messages

You control who receives email messages, the subject line, and some text in the body of the email. The body of the email always contains information about the alarm. This information is not editable. You can also control whether an email is sent based on severity levels. You can set different distribution lists for different rules and different severity levels, or set the same notification policy for all rules.

To configure email notifications:

1. In the Alarms view, click the Settings tab.
2. Decide on the scope for the change: all alarms, a category of rules, or a selected rule.
3. Complete the steps for the selected scope:
   - All alarms - Click All Alarms. Click the Define Email Settings button.
   - Category of rules - Click a category. Click the Define Email Settings button.
   - Selected rule - Click a rule. Click the Email Notification Settings tab.
4. If you selected All Alarms or a category, in the Email Notification Settings dialog box, do one of the following:
   - To change the severity levels that warrant an email notification, from the Messages will be enabled for severities box, select the desired levels of severity.
   - To configure the same email recipients and message for all severity levels, click Configure mail recipients for all Severities and then click All severities.
   - To configure different email recipients and messages for each of the severity levels, click Configure mail recipients for the following options and then click a severity level.
5. In the Message Settings dialog box, configure the email recipients and message. Note that you can use registry variables in place of email addresses. Type the variable name between two hash (#) symbols, for example: #EmailTeamName#. For more information, see Defining variables to contain email recipients.
   - To — Type the addresses of the people who need to take action when this alarm triggers.
   - CC — Type the addresses of the people who want to be notified when the alarm triggers.
   - Subject — Optional. Edit the text of the subject line to better suit your environment. Avoid editing the variables, which are identified with the @ symbol.
   - Body Prefix — Optional. Add text that should appear above the alarm information in the body of the email.
6. Click Set to save the message configuration and close the dialog box.
7. If the Edit Notification Settings dialog box is open, click Set.

Defining variables to contain email recipients

You can create registry variables that contain one or more email addresses, and use these registry variables when defining email notifications. This procedure describes how to create a registry value.

To create a registry variable:

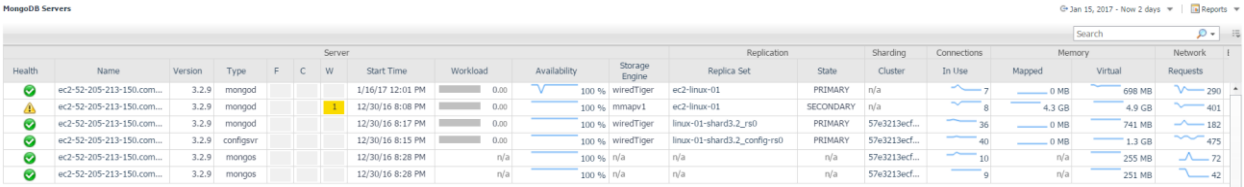
1. On the navigation panel, under Dashboards, click Administration > Rules & Notifications > Manage Registry Variables.
2. Click Add. The New Registry Variable Wizard opens.

3. Select the registry variable type String, and click Next.
4. In the Name field, enter a name, for example: EmailTeamName
5. Click Next.
6. Select Static Value.
7. In the Enter desired value box, enter one or more email addresses (separated by commas).
8. Click Finish.

Dashboards

MongoDB Servers

This top-level dashboard lists all monitored MongoDB Servers and contains important configuration information, alarm status, and key metrics for the server. The workload metric is used for comparing the amount of work a server is doing relative to another MongoDB server. Clicking the DB server's name will drill down into the Server Overview dashboard. If using the Infrastructure cartridge to monitor the host, clicking or hovering on the host name will provide more information on the DB server's host.

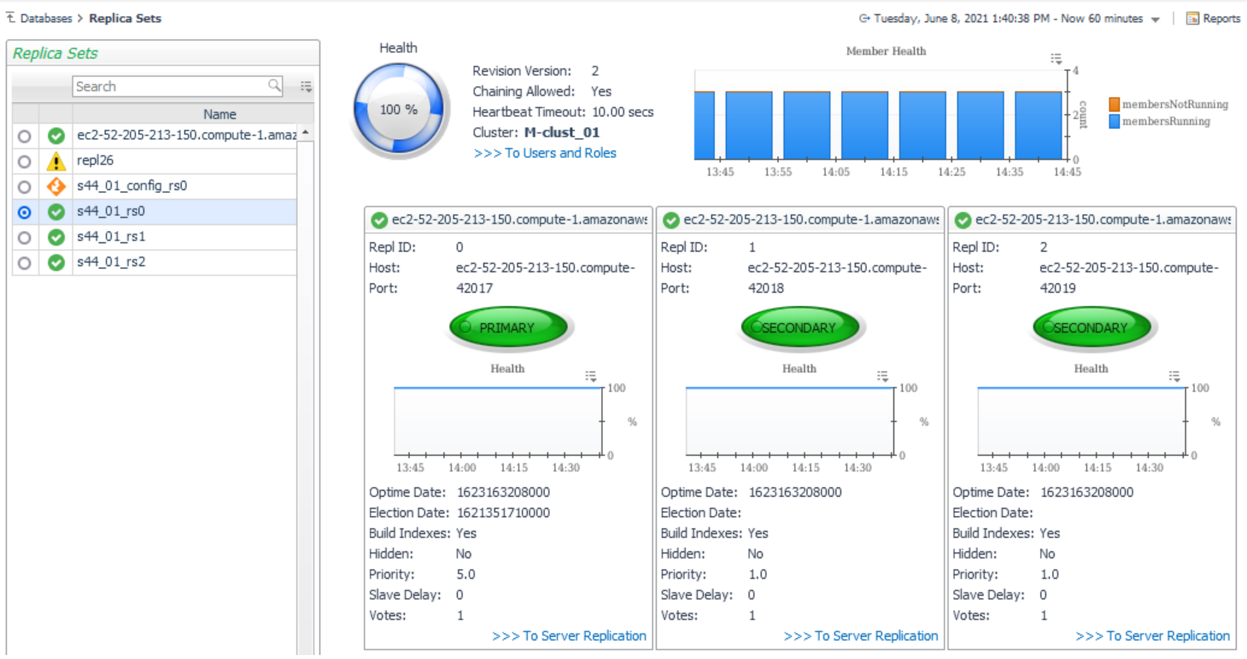


MongoDB Servers

	Server										Replication		Sharding	Connections	Memory		Network
Health	Name	Version	Type	F	C	W	Start Time	Workload	Availability	Storage Engine	Replica Set	State	Cluster	In Use	Mapped	Virtual	Requests
✓	ec2-52-205-213-150.com...	3.2.9	mongod				1/16/17 12:01 PM	0.00	100 %	wiredTiger	ec2-linux-01	PRIMARY	n/a	7	0 MB	698 MB	290
⚠	ec2-52-205-213-150.com...	3.2.9	mongod			1	12/30/16 8:08 PM	0.00	100 %	mmapv1	ec2-linux-01	SECONDARY	n/a	8	4.3 GB	4.9 GB	401
✓	ec2-52-205-213-150.com...	3.2.9	mongod				12/30/16 8:17 PM	0.00	100 %	wiredTiger	linux-01-shard3.2_rs0	PRIMARY	57e3213ecf...	36	0 MB	741 MB	182
✓	ec2-52-205-213-150.com...	3.2.9	configsvr				12/30/16 8:15 PM	0.00	100 %	wiredTiger	linux-01-shard3.2_config-rs0	PRIMARY	57e3213ecf...	40	0 MB	1.3 GB	475
✓	ec2-52-205-213-150.com...	3.2.9	mongos				12/30/16 8:28 PM	n/a	100 %	n/a	n/a	n/a	57e3213ecf...	10	n/a	255 MB	72
✓	ec2-52-205-213-150.com...	3.2.9	mongos				12/30/16 8:28 PM	n/a	100 %	n/a	n/a	n/a	57e3213ecf...	9	n/a	251 MB	42

Replica Sets

This top-level dashboard is dynamic and expands to provide a list of discovered Replica Sets. Selecting a specific Replica Set from the left-hand navigation pane automatically updates the right side of the page showing all members of the Replica Set. The Health indicator shows percentage of members currently running. The adjacent bar graph shows number of members running/not running. If the server is actively monitored by a MongoDB Agent, a link "Click for More Information" will appear, leading to the Replication page for that server, covered later in this document.



	Health	Name
○	✓	ec2-52-205-213-150.compute-1.amaz
○	⚠	repl26
○	◆	s44_01_config_rs0
●	✓	s44_01_rs0
○	✓	s44_01_rs1
○	✓	s44_01_rs2

Revision Version: 2
Chaining Allowed: Yes
Heartbeat Timeout: 10.00 secs
Cluster: **M-clust_01**
>>> To Users and Roles

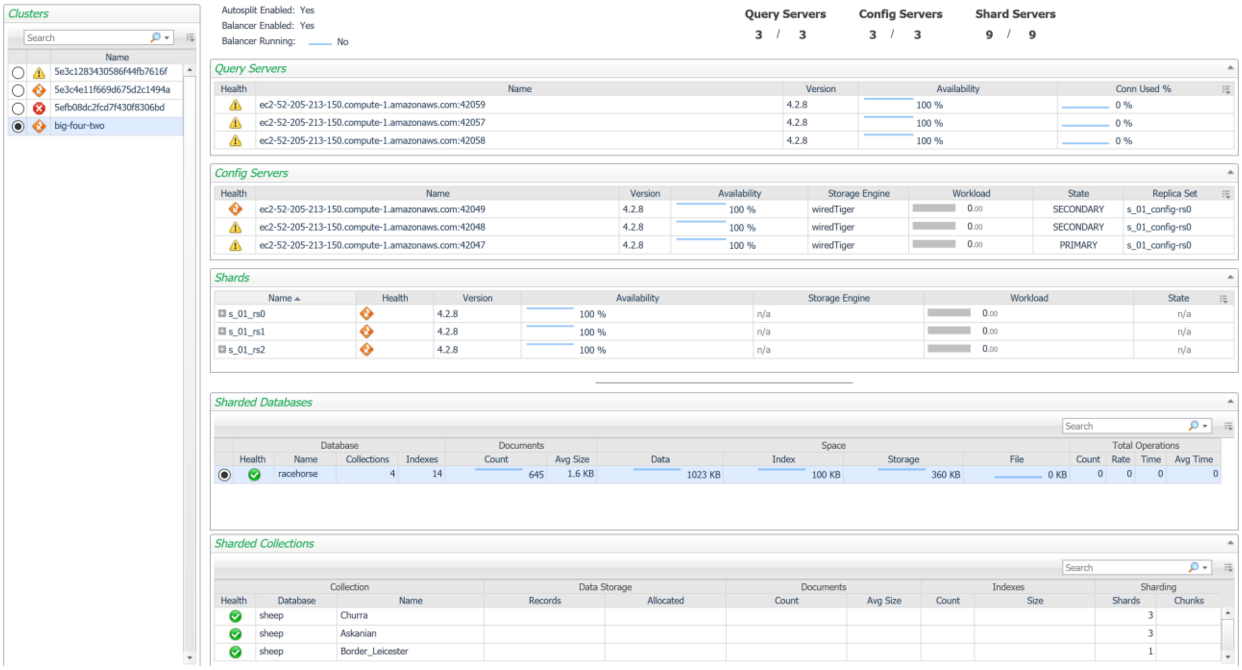
ec2-52-205-213-150.compute-1.amazonaws
Repl ID: 0
Host: ec2-52-205-213-150.compute-42017
Port:
Optime Date: 1623163208000
Election Date: 1621351710000
Build Indexes: Yes
Hidden: No
Priority: 5.0
Slave Delay: 0
Votes: 1
>>> To Server Replication

ec2-52-205-213-150.compute-1.amazonaws
Repl ID: 1
Host: ec2-52-205-213-150.compute-42018
Port:
Optime Date: 1623163208000
Election Date:
Build Indexes: Yes
Hidden: No
Priority: 1.0
Slave Delay: 0
Votes: 1
>>> To Server Replication

ec2-52-205-213-150.compute-1.amazonaws
Repl ID: 2
Host: ec2-52-205-213-150.compute-42019
Port:
Optime Date: 1623163208000
Election Date:
Build Indexes: Yes
Hidden: No
Priority: 1.0
Slave Delay: 0
Votes: 1
>>> To Server Replication

MongoDB Clusters

This top-level dashboard lists all monitored MongoDB clusters. Selecting a cluster updates the page with summary data on the cluster status as well as listing the different components in the cluster, including mongos, configsvr, shards, and databases and collections.



Clusters

		Name
○	⚠	5e3c1283430586f44fb7616f
○	◆	5e3c4e11f669d675d2c1494a
○	✖	5efb08dc2fcd7f430f8306bd
●	◆	big-four-two

Autosplit Enabled: Yes
Balancer Enabled: Yes
Balancer Running: No

Query Servers	Config Servers	Shard Servers
3 / 3	3 / 3	9 / 9

Query Servers

Health	Name	Version	Availability	Conn Used %
⚠	ec2-52-205-213-150.compute-1.amazonaws.com:42059	4.2.8	100 %	0 %
⚠	ec2-52-205-213-150.compute-1.amazonaws.com:42057	4.2.8	100 %	0 %
⚠	ec2-52-205-213-150.compute-1.amazonaws.com:42058	4.2.8	100 %	0 %

Config Servers

Health	Name	Version	Availability	Storage Engine	Workload	State	Replica Set
◆	ec2-52-205-213-150.compute-1.amazonaws.com:42049	4.2.8	100 %	wiredTiger	0.00	SECONDARY	s_01_config-rs0
⚠	ec2-52-205-213-150.compute-1.amazonaws.com:42048	4.2.8	100 %	wiredTiger	0.00	SECONDARY	s_01_config-rs0
⚠	ec2-52-205-213-150.compute-1.amazonaws.com:42047	4.2.8	100 %	wiredTiger	0.00	PRIMARY	s_01_config-rs0

Shards

Name ▲	Health	Version	Availability	Storage Engine	Workload	State
⊞ s_01_rs0	◆	4.2.8	100 %	n/a	0.00	n/a
⊞ s_01_rs1	◆	4.2.8	100 %	n/a	0.00	n/a
⊞ s_01_rs2	◆	4.2.8	100 %	n/a	0.00	n/a

Sharded Databases

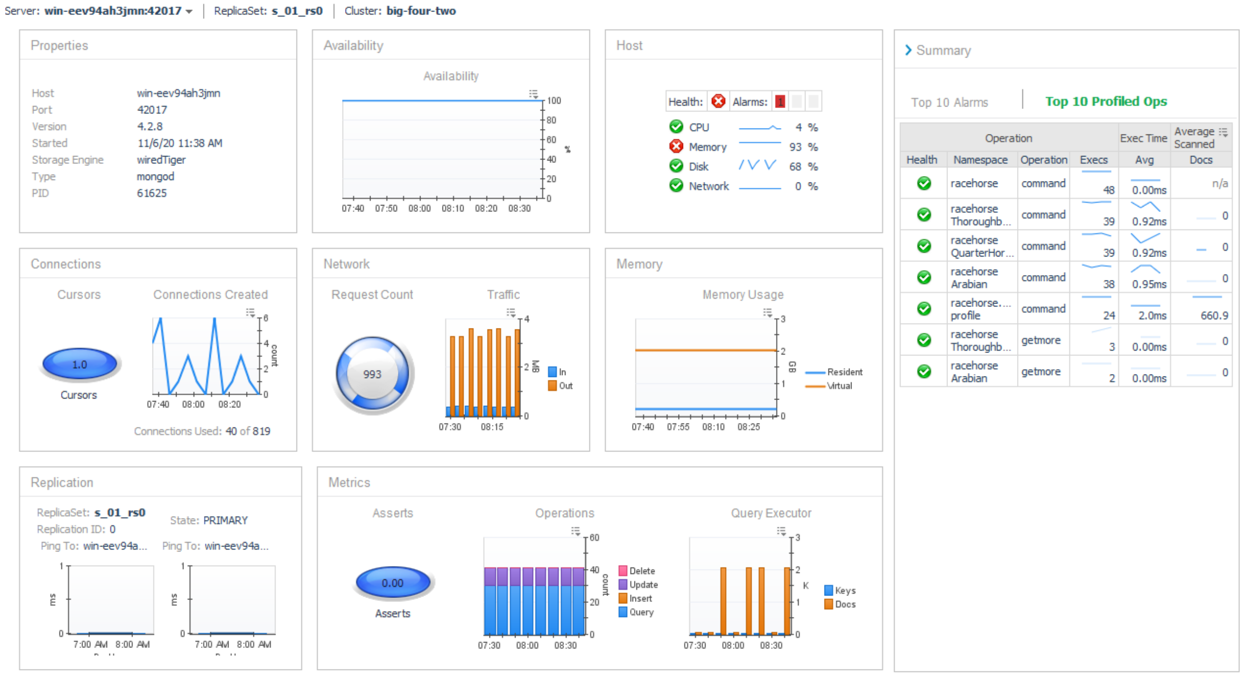
	Database				Documents		Space				Total Operations			
	Health	Name	Collections	Indexes	Count	Avg Size	Data	Index	Storage	File	Count	Rate	Time	Avg Time
●	✓	racehorse	4	14	645	1.6 KB	1023 KB	100 KB	360 KB	0 KB	0	0	0	0

Sharded Collections

Collection			Data Storage		Documents		Indexes		Sharding	
Health	Database	Name	Records	Allocated	Count	Avg Size	Count	Size	Shards	Chunks
✓	sheep	Churra							3	
✓	sheep	Askanian							3	
✓	sheep	Border_Leicester							1	

Server Overview

This dashboard provides a comprehensive view of the MongoDB Server, Network, and Host metrics. Operators can hover over titles and the dashboard will display the current state and any associated alarms. The Server Navigation Bar at top can be used to navigate to other pages containing more detailed information on this server. Also, the MongoDB Server Selector link in the action panel on the right of the screen exists for all pages at this level, allowing you to switch between servers without returning to the MongoDB Servers dashboard.



Server: **win-eev94ah3jmn:42017** | ReplicaSet: **s_01_rs0** | Cluster: **big-four-two**

Properties

Host	win-eev94ah3jmn
Port	42017
Version	4.2.8
Started	11/6/20 11:38 AM
Storage Engine	wiredTiger
Type	mongod
PID	61625

Host

Health: ✗ Alarms: 1

✓	CPU	4 %
✗	Memory	93 %
✓	Disk	68 %
✓	Network	0 %

Connections Used: 40 of 819

Replication

ReplicaSet: **s_01_rs0**
Replication ID: 0
State: PRIMARY
Ping To: win-eev94a...
Ping To: win-eev94a...

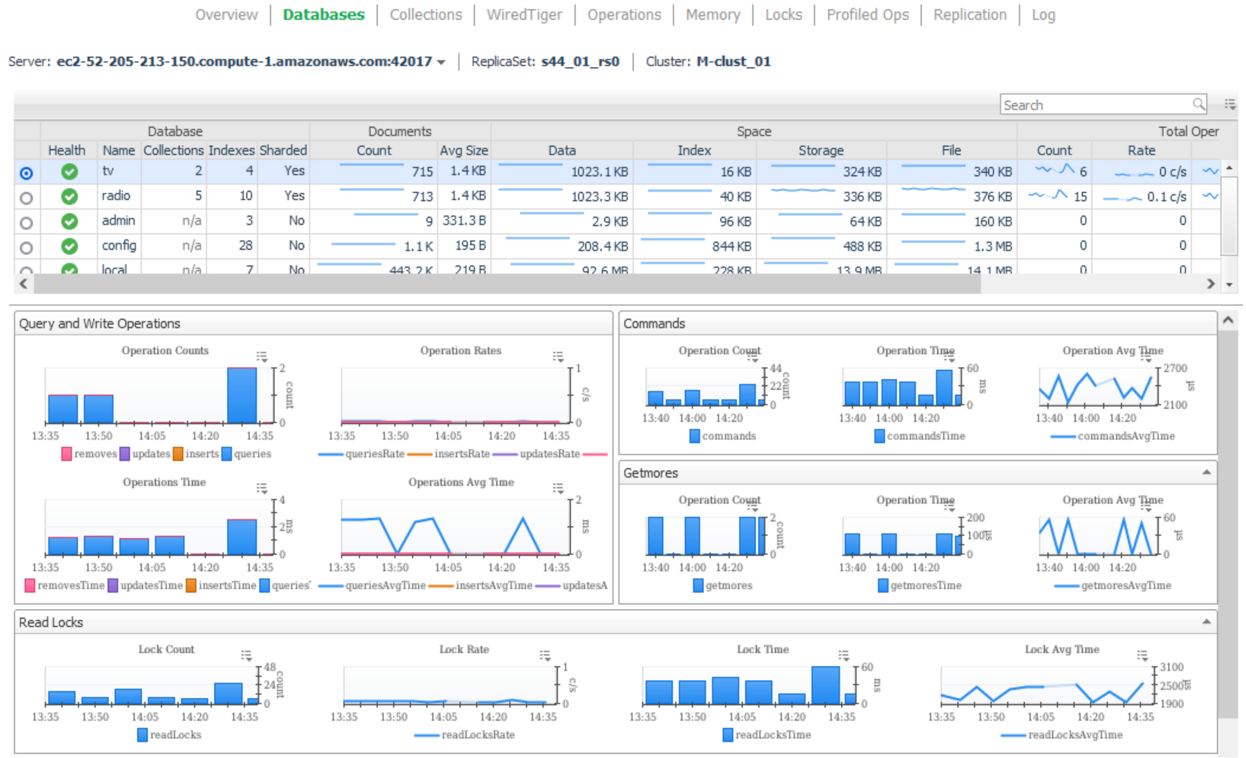
Summary

Top 10 Alarms | **Top 10 Profiled Ops**

Operation				Exec Time	Average Scanned
Health	Namespace	Operation	Execs	Avg	Docs
✓	racehorse	command	48	0.00ms	n/a
✓	racehorse Thoroughb...	command	39	0.92ms	0
✓	racehorse QuarterHor...	command	39	0.92ms	0
✓	racehorse Arabian	command	38	0.95ms	0
✓	racehorse.... profile	command	24	2.0ms	660.9
✓	racehorse Thoroughb...	getmore	3	0.00ms	0
✓	racehorse Arabian	getmore	2	0.00ms	0

Databases

The Databases dashboard lists all non-system databases for the server, along with key information. Choosing a row will update the bottom section to show Operation statistics for the selected database. Clicking a value in the Collections column will drill down to the Collections page for this server with only the collections for the database row shown by default.



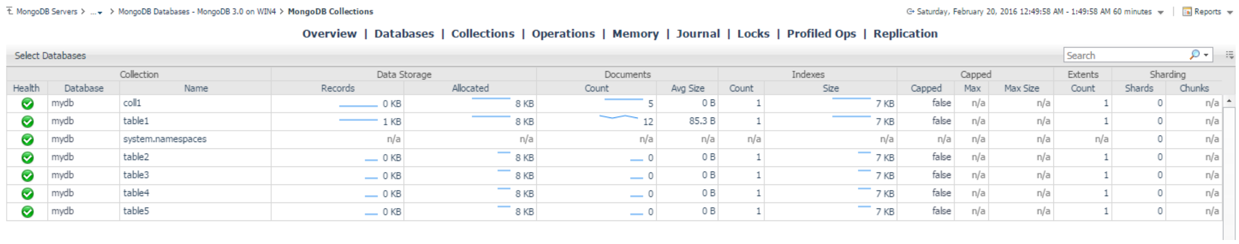
Overview | **Databases** | Collections | WiredTiger | Operations | Memory | Locks | Profiled Ops | Replication | Log

Server: **ec2-52-205-213-150.compute-1.amazonaws.com:42017** ▾ | ReplicaSet: **s44_01_rs0** | Cluster: **M-clust_01**

	Database					Documents		Space				Total Oper	
	Health	Name	Collections	Indexes	Sharded	Count	Avg Size	Data	Index	Storage	File	Count	Rate
●	✓	tv	2	4	Yes	715	1.4 KB	1023.1 KB	16 KB	324 KB	340 KB	6	0 c/s
○	✓	radio	5	10	Yes	713	1.4 KB	1023.3 KB	40 KB	336 KB	376 KB	15	0.1 c/s
○	✓	admin	n/a	3	No	9	331.3 B	2.9 KB	96 KB	64 KB	160 KB	0	0
○	✓	config	n/a	28	No	1.1 K	195 B	208.4 KB	844 KB	488 KB	1.3 MB	0	0
○	✓	local	n/a	7	No	443.2 K	219 B	92.6 MB	228 KB	13.9 MB	14.1 MB	0	0

Collections

This dashboard lists non-system collections in the MongoDB Server with identifying information and size and configuration data. Only collections for the selected databases will be shown. You can change which database(s) is/are selected by using the Select Databases option at the top-left portion of the table or the action panel on the right of the window. Clicking a collection name will drill down to the Collection page, covered below.



MongoDB Servers > ... > MongoDB Databases - MongoDB 3.0 on WIN4 > **MongoDB Collections**

Saturday, February 20, 2016 12:49:58 AM - 1:49:58 AM 60 minutes ▾ | Reports ▾

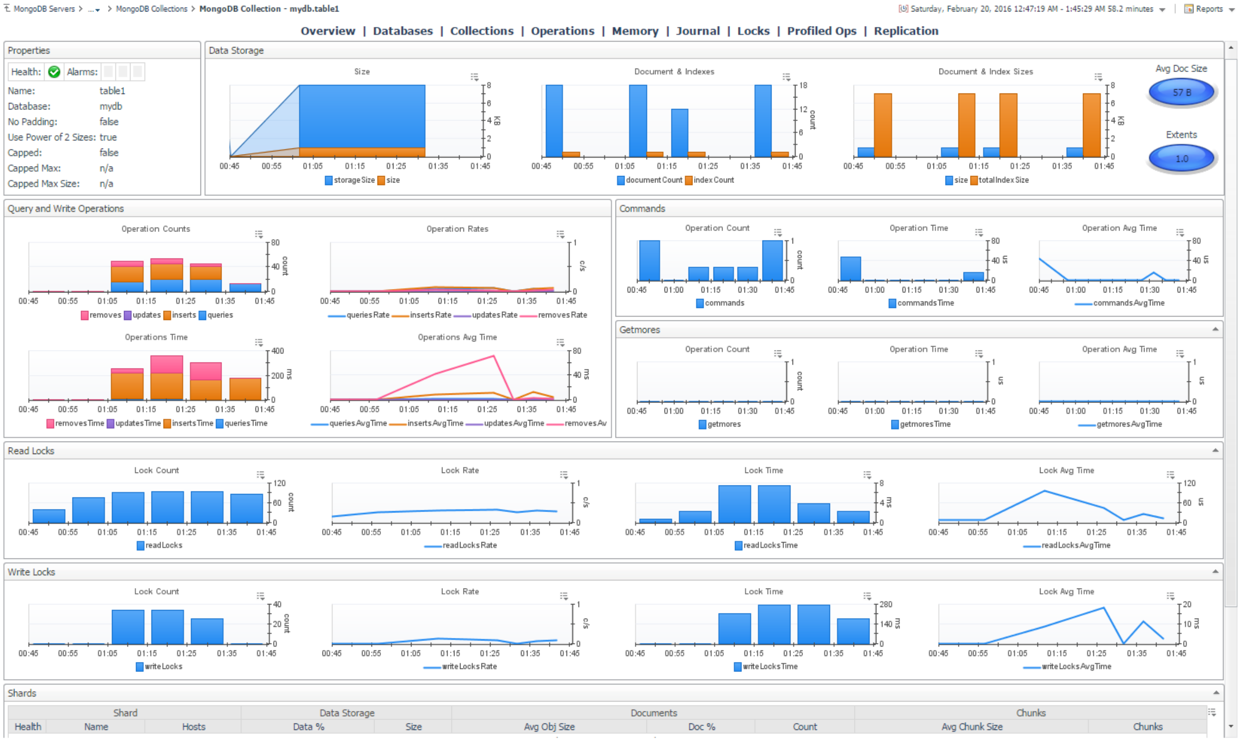
Overview | Databases | Collections | Operations | Memory | Journal | Locks | Profiled Ops | Replication

Select Databases

Collection			Data Storage		Documents		Indexes		Capped			Extents	Sharding	
Health	Database	Name	Records	Allocated	Count	Avg Size	Count	Size	Capped	Max	Max Size	Count	Shards	Chunks
✓	mydb	coll1	0 KB	8 KB	5	0 B	1	7 KB	false	n/a	n/a	1	0	n/a
✓	mydb	table1	1 KB	8 KB	12	85.3 B	1	7 KB	false	n/a	n/a	1	0	n/a
✓	mydb	system.namespaces	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
✓	mydb	table2	0 KB	8 KB	0	0 B	1	7 KB	false	n/a	n/a	1	0	n/a
✓	mydb	table3	0 KB	8 KB	0	0 B	1	7 KB	false	n/a	n/a	1	0	n/a
✓	mydb	table4	0 KB	8 KB	0	0 B	1	7 KB	false	n/a	n/a	1	0	n/a
✓	mydb	table5	0 KB	8 KB	0	0 B	1	7 KB	false	n/a	n/a	1	0	n/a

Collection Statistics

This page shows all available information for a collection, including properties, data storage information, operational metrics, and sharding, if enabled.



MongoDB Servers > ... > MongoDB Collections > MongoDB Collection - mydb.table1

Saturday, February 20, 2016 12:47:19 AM - 1:45:29 AM 58.2 minutes | Reports

Overview | Databases | Collections | Operations | Memory | Journal | Locks | Profiled Ops | Replication

Properties

Health: ✓ Alarms:

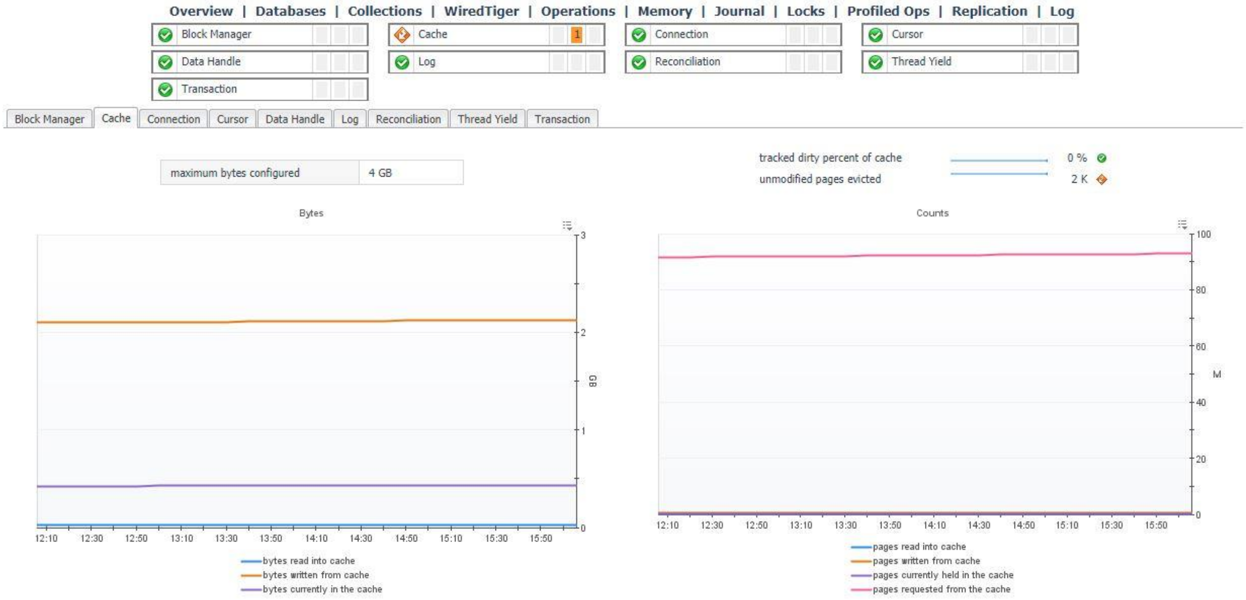
Name:	table1
Database:	mydb
No Padding:	false
Use Power of 2 Sizes:	true
Capped:	false
Capped Max:	n/a
Capped Max Size:	n/a

Shards

Shard			Data Storage		Documents			Chunks	
Health	Name	Hosts	Data %	Size	Avg Obj Size	Doc %	Count	Avg Chunk Size	Chunks

WiredTiger

This dashboard exposes the technical inner workings of the WiredTiger storage engine. Metrics include cache size and state, transaction ticket queues, memory and IO activity, cursor and session counts, and log size and operations.



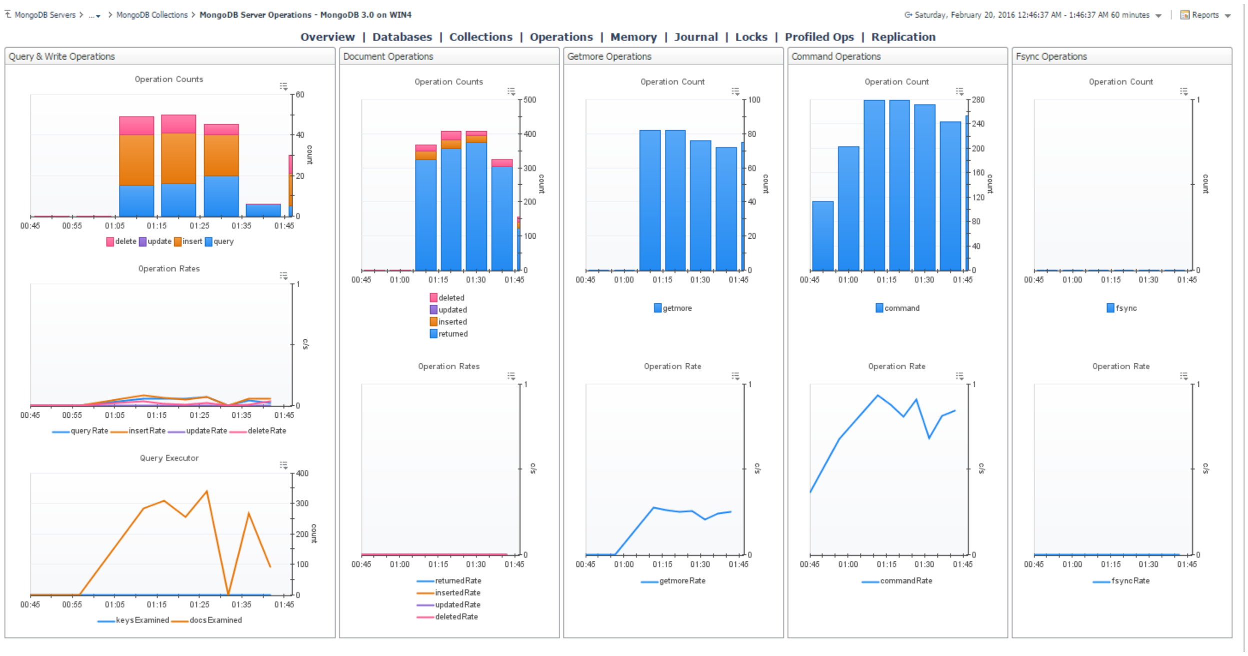
maximum bytes configured 4 GB

tracked dirty percent of cache 0 %

unmodified pages evicted 2 K

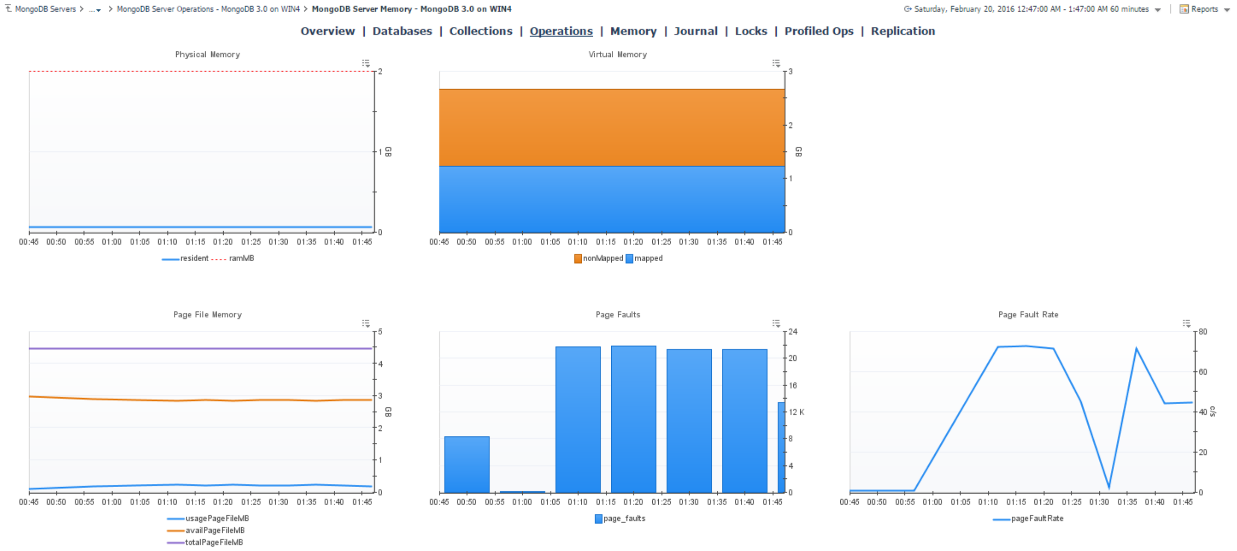
Operations

The operations dashboard updates based on the agent properties setting but can be adjusted to near-real-time monitoring and set to automatically refresh every 60 Seconds (No manual Refresh is required). The dashboard shows operations counts for Query & Write Operations, Document Operations, Getmore Operations, Command Operations, and Fsync Operations.



Memory

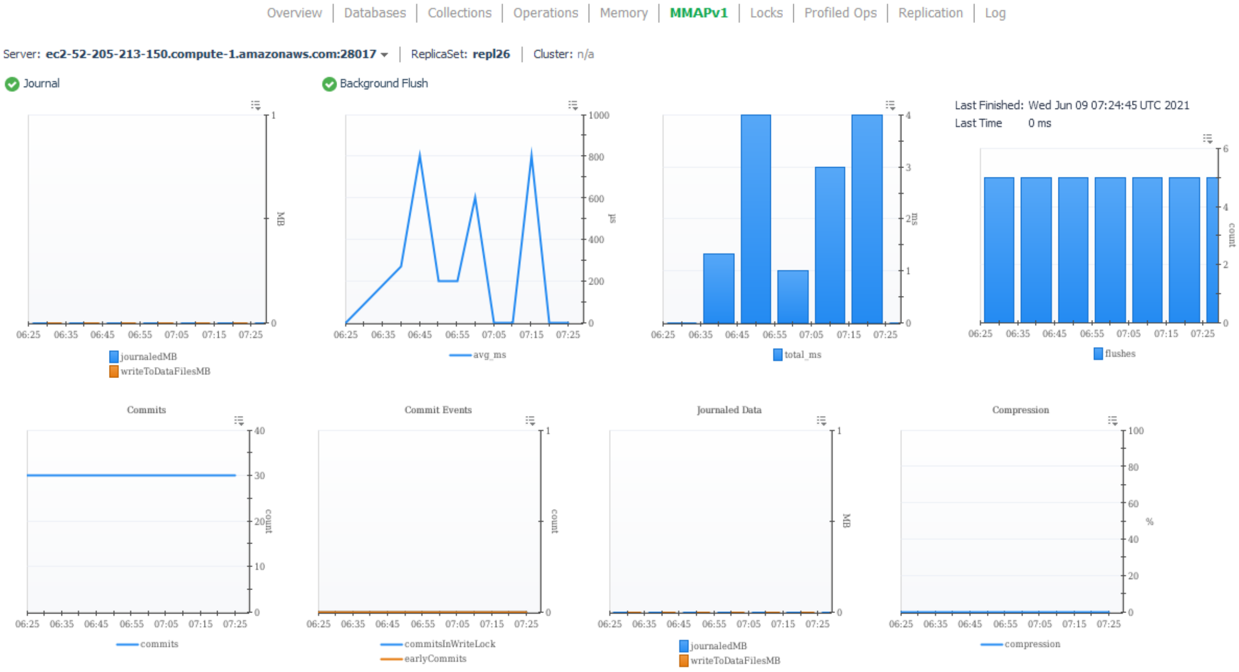
The memory dashboard displays metrics for Physical and Virtual Memory in addition to Page File Memory, and the number of Page Faults and the Page Fault Rate.


MongoDB Servers > … > MongoDB Server Operations - MongoDB 3.0 on WIN4 > **MongoDB Server Memory - MongoDB 3.0 on WIN4**

Saturday, February 20, 2016 12:47:00 AM - 1:47:00 AM 60 minutes

MMAPv1

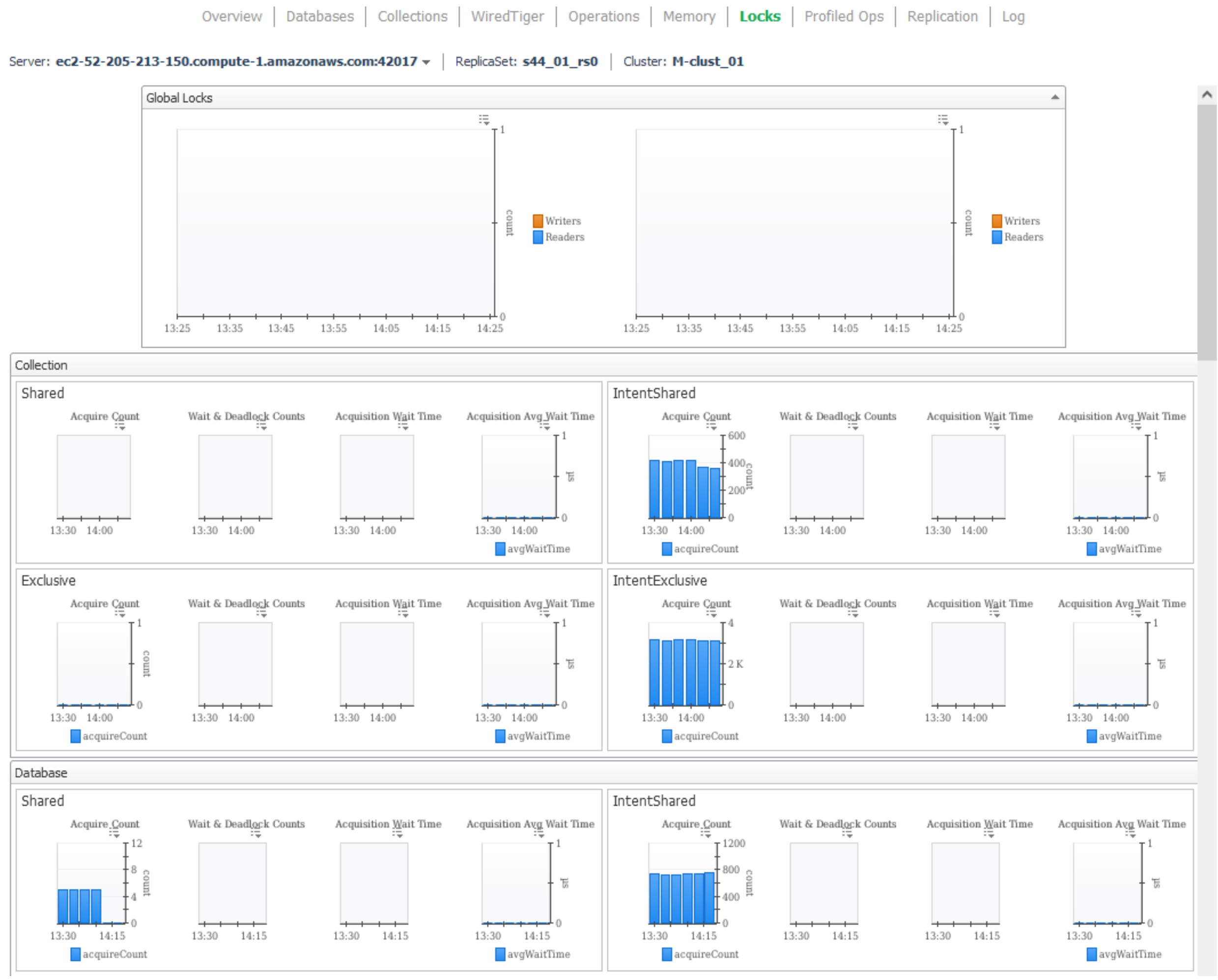
The MMAPv1 Dashboard shows the number of Background Flush, Commits, Commit Events, Journaled Data, Compression Percentage, Commit Timings and Journal Timings for the MMAPv1 journal, if it is enabled. A dashboard for monitoring the equivalent WiredTiger feature can be found in the Log tab on the WiredTiger dashboard.


Server: **ec2-52-205-213-150.compute-1.amazonaws.com:28017** | ReplicaSet: **repl26** | Cluster: n/a

Last Finished: Wed Jun 09 07:24:45 UTC 2021
Last Time 0 ms

Locks

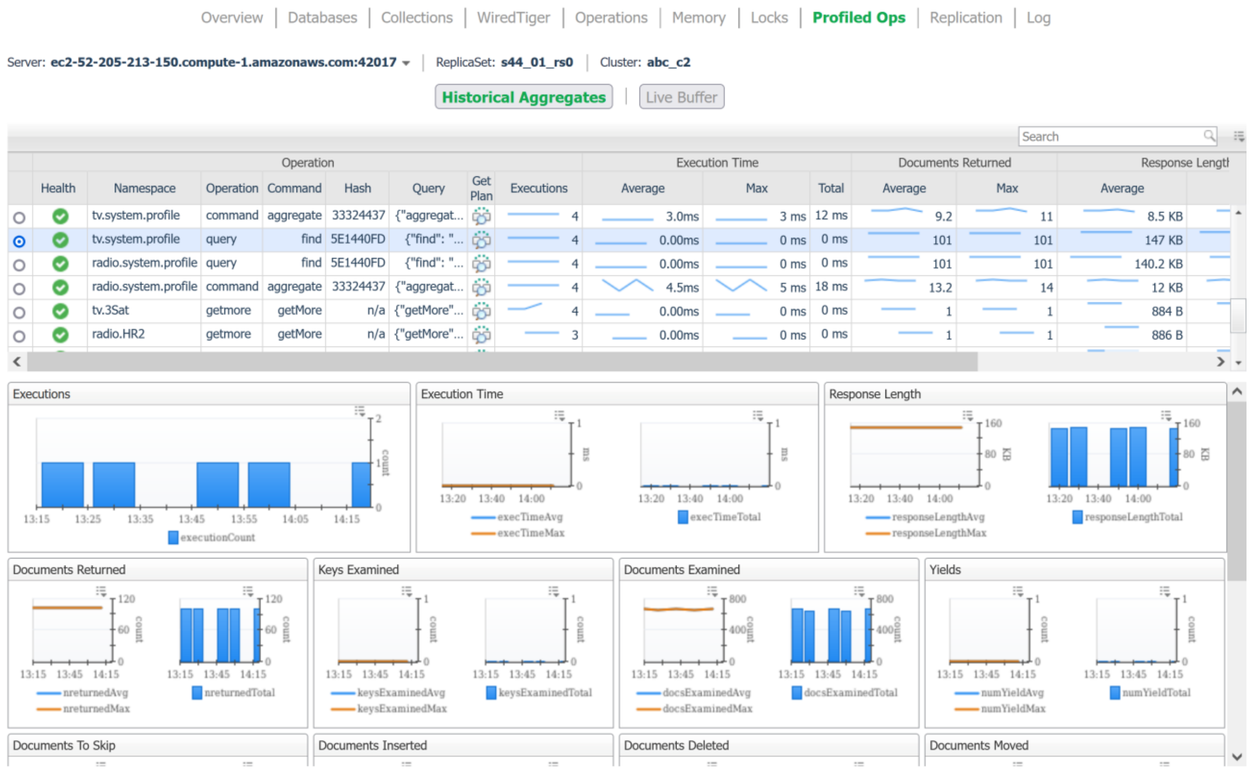
The Locks Dashboard displays a dynamic collection of locks for various lock types. MongoDB collects information on locks at a number of levels within its architecture. This dashboard will capture all lock information present. Examples of lock types include: collection; database; global, and oplog. For each type, metrics are shown for shared, intent shared, exclusive, and intent exclusive locks.



Server: ec2-52-205-213-150.compute-1.amazonaws.com:42017 | ReplicaSet: s44_01_rs0 | Cluster: M-clust_01

Profiled Operations

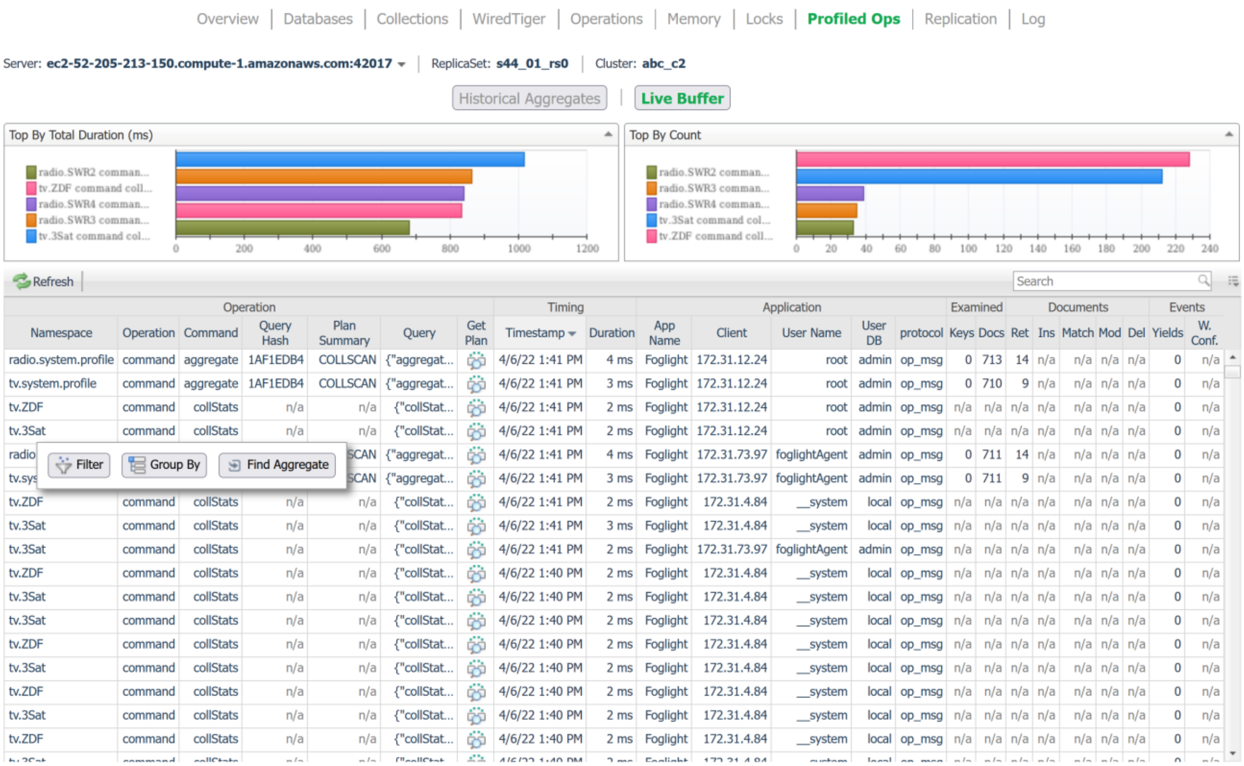
This dashboard has two main views: the Historical Aggregates view and the Live Buffer view. The Historical Aggregates view shows collected profiled operations that have been aggregated into groups for statistical analysis. System profiling must be enabled on the MongoDB server. Profiled operations are collected for a maximum of 20 databases. Metrics shown in the table are for the selected period (the time window selected at the top-right of the page). The page shows general data on each operation, such as executions and execution time, documents returned, response length, and scanned objects as well as information specific to the type of the selected operation.



Server: ec2-52-205-213-150.compute-1.amazonaws.com:42017 | ReplicaSet: s44_01_rs0 | Cluster: abc_c2

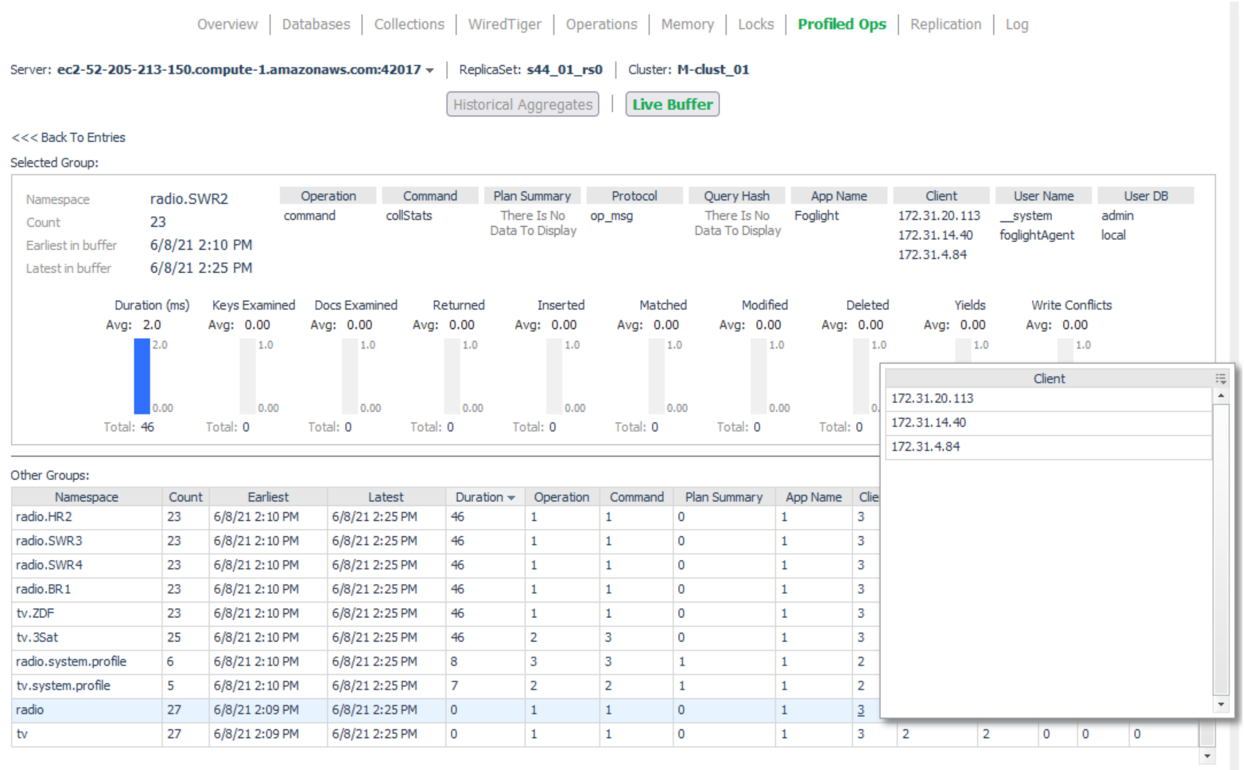
	Operation								Execution Time			Documents Returned		Response Lengt
	Health	Namespace	Operation	Command	Hash	Query	Get Plan	Executions	Average	Max	Total	Average	Max	Average
○	✓	tv.system.profile	command	aggregate	33324437	{"aggregat...		4	3.0ms	3 ms	12 ms	9.2	11	8.5 KB
●	✓	tv.system.profile	query	find	5E1440FD	{"find": "...		4	0.00ms	0 ms	0 ms	101	101	147 KB
○	✓	radio.system.profile	query	find	5E1440FD	{"find": "...		4	0.00ms	0 ms	0 ms	101	101	140.2 KB
○	✓	radio.system.profile	command	aggregate	33324437	{"aggregat...		4	4.5ms	5 ms	18 ms	13.2	14	12 KB
○	✓	tv.3Sat	getmore	getMore	n/a	{"getMore"...		4	0.00ms	0 ms	0 ms	1	1	884 B
○	✓	radio.HR2	getmore	getMore	n/a	{"getMore"...		3	0.00ms	0 ms	0 ms	1	1	886 B

The Live Buffer view shows the details of individual profiled operations currently visible on the MongoDB instance. The top 5 operations by total duration and by count are displayed, along with a list of current operations. Three actions are available for any non-numerical property: Filter, Group By, and Find Aggregate. Additionally, the live buffer entries can be updated by clicking the Refresh button. The Filter action updates the table to only show operations with the selected value of the respective property (table column). Multiple columns can be filtered sequentially.



Operation							Timing		Application					Examined		Documents					Events	
Namespace	Operation	Command	Query Hash	Plan Summary	Query	Get Plan	Timestamp	Duration	App Name	Client	User Name	User DB	protocol	Keys	Docs	Ret	Ins	Match	Mod	Del	Yields	W. Conf.
radio.system.profile	command	aggregate	1AF1EDB4	COLLSCAN	{"aggregat...		4/6/22 1:41 PM	4 ms	Foglight	172.31.12.24	root	admin	op_msg	0	713	14	n/a	n/a	n/a	n/a	0	n/a
tv.system.profile	command	aggregate	1AF1EDB4	COLLSCAN	{"aggregat...		4/6/22 1:41 PM	3 ms	Foglight	172.31.12.24	root	admin	op_msg	0	710	9	n/a	n/a	n/a	n/a	0	n/a
tv.ZDF	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:41 PM	2 ms	Foglight	172.31.12.24	root	admin	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.3Sat	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:41 PM	2 ms	Foglight	172.31.12.24	root	admin	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
radio				SCAN	{"aggregat...		4/6/22 1:41 PM	4 ms	Foglight	172.31.73.97	foglightAgent	admin	op_msg	0	711	14	n/a	n/a	n/a	n/a	0	n/a
tv.sys				SCAN	{"aggregat...		4/6/22 1:41 PM	3 ms	Foglight	172.31.73.97	foglightAgent	admin	op_msg	0	711	9	n/a	n/a	n/a	n/a	0	n/a
tv.ZDF	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:41 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.3Sat	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:41 PM	3 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.3Sat	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:41 PM	2 ms	Foglight	172.31.73.97	foglightAgent	admin	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.ZDF	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.3Sat	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.3Sat	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.ZDF	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.3Sat	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.ZDF	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.3Sat	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a
tv.ZDF	command	collStats	n/a	n/a	{"collStat...		4/6/22 1:40 PM	2 ms	Foglight	172.31.4.84	__system	local	op_msg	n/a	n/a	n/a	n/a	n/a	n/a	n/a	0	n/a

Group By aggregates the entries in the live buffer according to the selected property (table column). Unique values for other properties are presented, along with aggregated numeric values. One group is shown in detail at the top. Other groups may be moved to the detail view by selecting the respective value of the property grouped by (the first column of the "Other Groups" table.



Overview | Databases | Collections | WiredTiger | Operations | Memory | Locks | Profiled Ops | Replication | Log
Historical Aggregates | Live Buffer
<<< Back To Entries

Server: ec2-52-205-213-150.compute-1.amazonaws.com:42017 | ReplicaSet: s44_01_rs0 | Cluster: M-clust_01

Selected Group:

Namespace	radio.SWR2
Count	23
Earliest in buffer	6/8/21 2:10 PM
Latest in buffer	6/8/21 2:25 PM

Operation	Command	Plan Summary	Protocol	Query Hash	App Name	Client	User Name	User DB
command	collStats	There Is No Data To Display	op_msg	There Is No Data To Display	Foglight	172.31.20.113 172.31.14.40 172.31.4.84	__system foglightAgent	admin local

Duration (ms)	Keys Examined	Docs Examined	Returned	Inserted	Matched	Modified	Deleted	Yields	Write Conflicts
Avg: 2.0	Avg: 0.00	Avg: 0.00	Avg: 0.00	Avg: 0.00	Avg: 0.00	Avg: 0.00	Avg: 0.00	Avg: 0.00	Avg: 0.00
Total: 46	Total: 0	Total: 0	Total: 0	Total: 0	Total: 0	Total: 0	Total: 0		

Other Groups:

Namespace	Count	Earliest	Latest	Duration	Operation	Command	Plan Summary	App Name	Clie					
radio.HR2	23	6/8/21 2:10 PM	6/8/21 2:25 PM	46	1	1	0	1	3					
radio.SWR3	23	6/8/21 2:10 PM	6/8/21 2:25 PM	46	1	1	0	1	3					
radio.SWR4	23	6/8/21 2:10 PM	6/8/21 2:25 PM	46	1	1	0	1	3					
radio.BR1	23	6/8/21 2:10 PM	6/8/21 2:25 PM	46	1	1	0	1	3					
tv.ZDF	23	6/8/21 2:10 PM	6/8/21 2:25 PM	46	1	1	0	1	3					
tv.3Sat	25	6/8/21 2:10 PM	6/8/21 2:25 PM	46	2	3	0	1	3					
radio.system.profile	6	6/8/21 2:10 PM	6/8/21 2:25 PM	8	3	3	1	1	2					
tv.system.profile	5	6/8/21 2:10 PM	6/8/21 2:25 PM	7	2	2	1	1	2					
radio	27	6/8/21 2:09 PM	6/8/21 2:25 PM	0	1	1	0	1	3					
tv	27	6/8/21 2:09 PM	6/8/21 2:25 PM	0	1	1	0	1	3	2	2	0	0	0

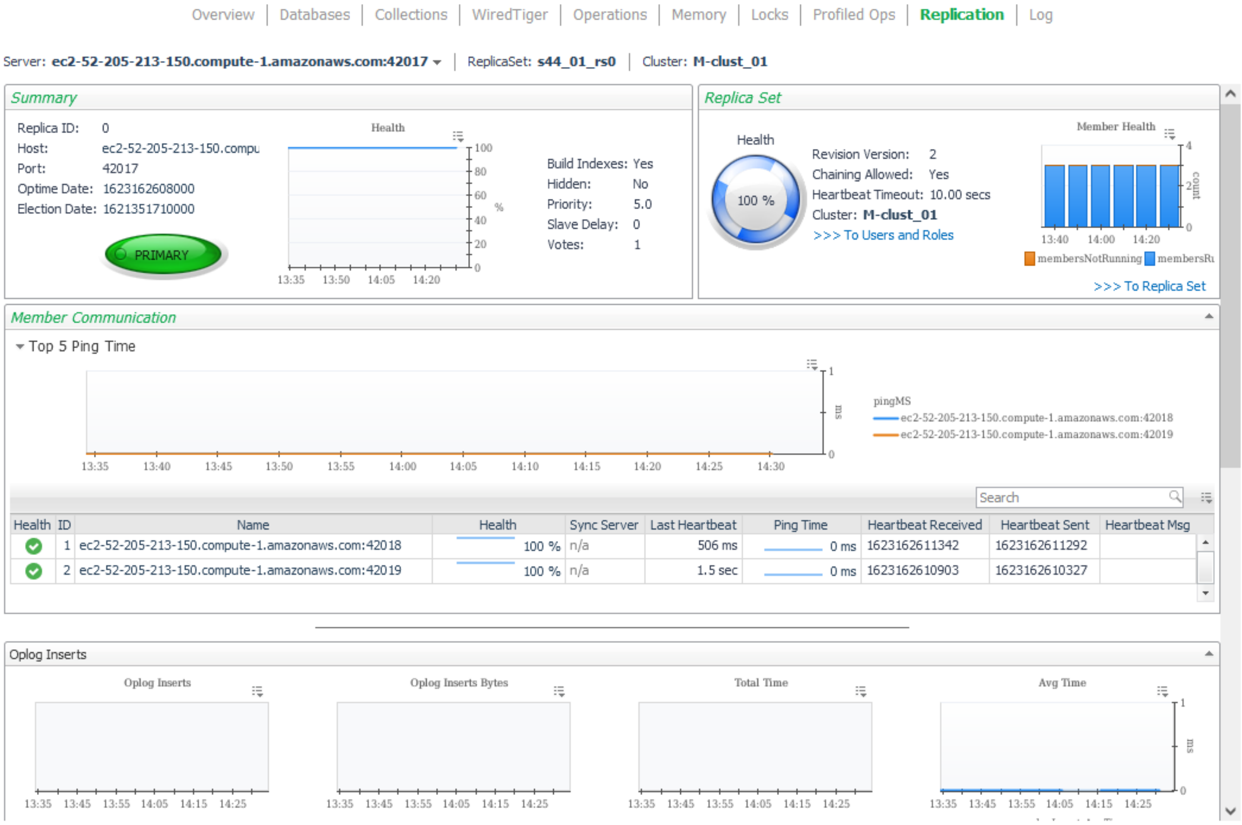
Client
172.31.20.113
172.31.14.40
172.31.4.84

The Group By action is performed on the currently visible live buffer entries. Hence it may be preceded by the Filter action to see an aggregated view of a filtered subset of the currently available data.

Finally, the Find Aggregate action switches to the Historical Aggregate view and selects the aggregated operation entry corresponding to the selected Live Buffer entry.

Replication

This dashboard shows Replication information for a MongoDB server. It contains summaries of the server status and replica set to which it belongs, followed by a section on Replica Members as seen from the current server’s perspective, followed by performance metrics covering operations used in the replication process.



Server: **ec2-52-205-213-150.compute-1.amazonaws.com:42017** | ReplicaSet: **s44_01_rs0** | Cluster: **M-clust_01**

Summary

Replica ID: 0
Host: ec2-52-205-213-150.compu
Port: 42017
Optime Date: 1623162608000
Election Date: 1621351710000

Build Indexes: Yes
Hidden: No
Priority: 5.0
Slave Delay: 0
Votes: 1

Replica Set

Revision Version: 2
Chaining Allowed: Yes
Heartbeat Timeout: 10.00 secs
Cluster: **M-clust_01**

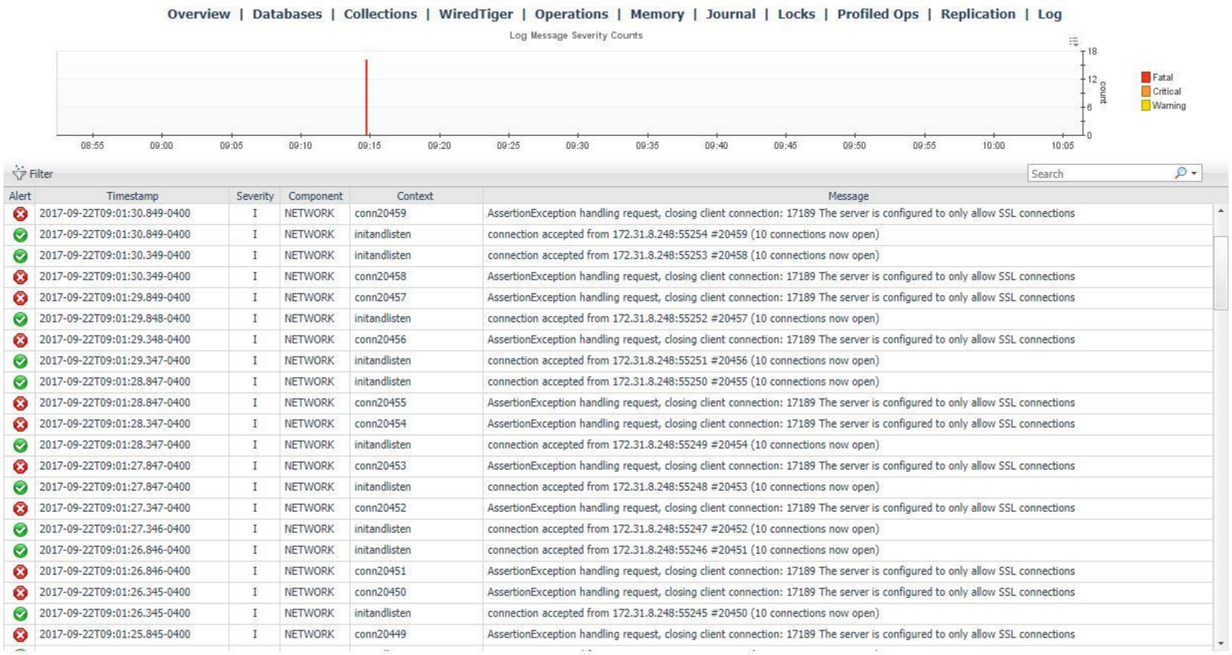
Member Communication

Health	ID	Name	Health	Sync Server	Last Heartbeat	Ping Time	Heartbeat Received	Heartbeat Sent	Heartbeat Msg
✓	1	ec2-52-205-213-150.compute-1.amazonaws.com:42018	100 %	n/a	506 ms	0 ms	1623162611342	1623162611292	
✓	2	ec2-52-205-213-150.compute-1.amazonaws.com:42019	100 %	n/a	1.5 sec	0 ms	1623162610903	1623162610327	

Oplog Inserts

Log

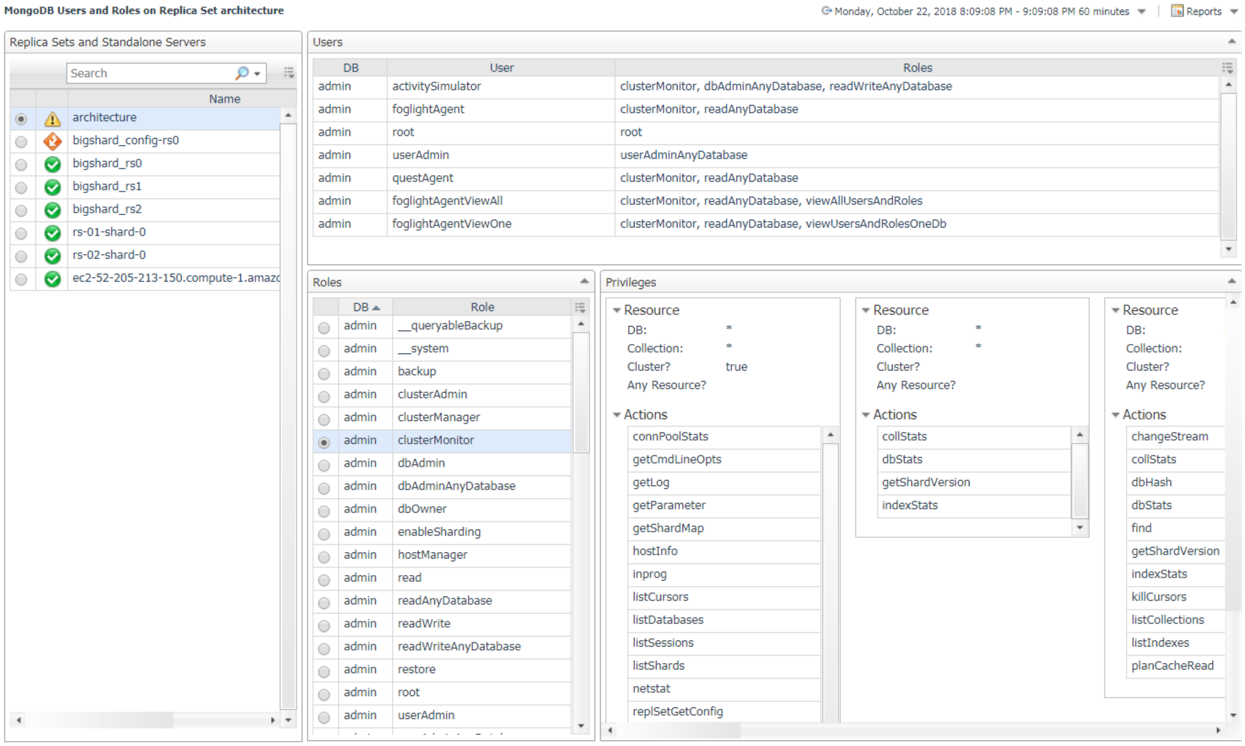
The log dashboard samples current log entries generated by the MongoDB process. Entries that have generated an alert are graphed above, while all alerts displayed can be filtered by log severity (Fatal, Debug, etc.) or log component (NETWORK, SHARDING, STORAGE, etc.).


Overview | Databases | Collections | WiredTiger | Operations | Memory | Journal | Locks | Profiled Ops | Replication | Log
Log Message Severity Counts
18
12
6
0
count
08:55
09:00
09:05
09:10
09:15
09:20
09:25
09:30
09:35
09:40
09:45
09:50
09:55
10:00
10:05
🟥 Fatal
🟧 Critical
🟨 Warning
Filter
Search

Alert	Timestamp	Severity	Component	Context	Message
✗	2017-09-22T09:01:30.849-0400	I	NETWORK	conn20459	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✓	2017-09-22T09:01:30.849-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55254 #20459 (10 connections now open)
✓	2017-09-22T09:01:30.349-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55253 #20458 (10 connections now open)
✗	2017-09-22T09:01:30.349-0400	I	NETWORK	conn20458	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✗	2017-09-22T09:01:29.849-0400	I	NETWORK	conn20457	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✓	2017-09-22T09:01:29.848-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55252 #20457 (10 connections now open)
✗	2017-09-22T09:01:29.348-0400	I	NETWORK	conn20456	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✓	2017-09-22T09:01:29.347-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55251 #20456 (10 connections now open)
✓	2017-09-22T09:01:28.847-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55250 #20455 (10 connections now open)
✗	2017-09-22T09:01:28.847-0400	I	NETWORK	conn20455	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✗	2017-09-22T09:01:28.347-0400	I	NETWORK	conn20454	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✓	2017-09-22T09:01:28.347-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55249 #20454 (10 connections now open)
✗	2017-09-22T09:01:27.847-0400	I	NETWORK	conn20453	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✓	2017-09-22T09:01:27.847-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55248 #20453 (10 connections now open)
✗	2017-09-22T09:01:27.347-0400	I	NETWORK	conn20452	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✓	2017-09-22T09:01:27.346-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55247 #20452 (10 connections now open)
✓	2017-09-22T09:01:26.846-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55246 #20451 (10 connections now open)
✗	2017-09-22T09:01:26.846-0400	I	NETWORK	conn20451	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✗	2017-09-22T09:01:26.345-0400	I	NETWORK	conn20450	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections
✓	2017-09-22T09:01:26.345-0400	I	NETWORK	initandlisten	connection accepted from 172.31.8.248:55245 #20450 (10 connections now open)
✗	2017-09-22T09:01:25.845-0400	I	NETWORK	conn20449	AssertionException handling request, closing client connection: 17189 The server is configured to only allow SSL connections

Users and Roles

The Users and Roles dashboard shows all users, roles, and privileges visible on the server. They are organized by the replica set or standalone instance on which they are found. Note that the users on a sharded cluster that are accessed through the mongos query servers are those that are found on the cluster's config server replica set. Select an individual role to see its associated actions.



MongoDB Users and Roles on Replica Set architecture

Monday, October 22, 2018 8:09:08 PM - 9:09:08 PM 60 minutes | Reports

Replica Sets and Standalone Servers

Search

		Name
●	⚠	architecture
○	◆	bigshard_config-rs0
○	✓	bigshard_rs0
○	✓	bigshard_rs1
○	✓	bigshard_rs2
○	✓	rs-01-shard-0
○	✓	rs-02-shard-0
○	✓	ec2-52-205-213-150.compute-1.amazo

Users

DB	User	Roles
admin	activitySimulator	clusterMonitor, dbAdminAnyDatabase, readWriteAnyDatabase
admin	foglightAgent	clusterMonitor, readAnyDatabase
admin	root	root
admin	userAdmin	userAdminAnyDatabase
admin	questAgent	clusterMonitor, readAnyDatabase
admin	foglightAgentViewAll	clusterMonitor, readAnyDatabase, viewAllUsersAndRoles
admin	foglightAgentViewOne	clusterMonitor, readAnyDatabase, viewUsersAndRolesOneDb

Roles

	DB ▲	Role
○	admin	__queryableBackup
○	admin	__system
○	admin	backup
○	admin	clusterAdmin
○	admin	clusterManager
●	admin	clusterMonitor
○	admin	dbAdmin
○	admin	dbAdminAnyDatabase
○	admin	dbOwner
○	admin	enableSharding
○	admin	hostManager
○	admin	read
○	admin	readAnyDatabase
○	admin	readWrite
○	admin	readWriteAnyDatabase
○	admin	restore
○	admin	root
○	admin	userAdmin

Privileges

Resource
DB: *
Collection: *
Cluster? true
Any Resource?

Actions
- connPoolStats
- getCmdLineOpts
- getLog
- getParameter
- getShardMap
- hostInfo
- inprog
- listCursors
- listDatabases
- listSessions
- listShards
- netstat
- replSetGetConfig

Resource
DB: *
Collection: *
Cluster?
Any Resource?

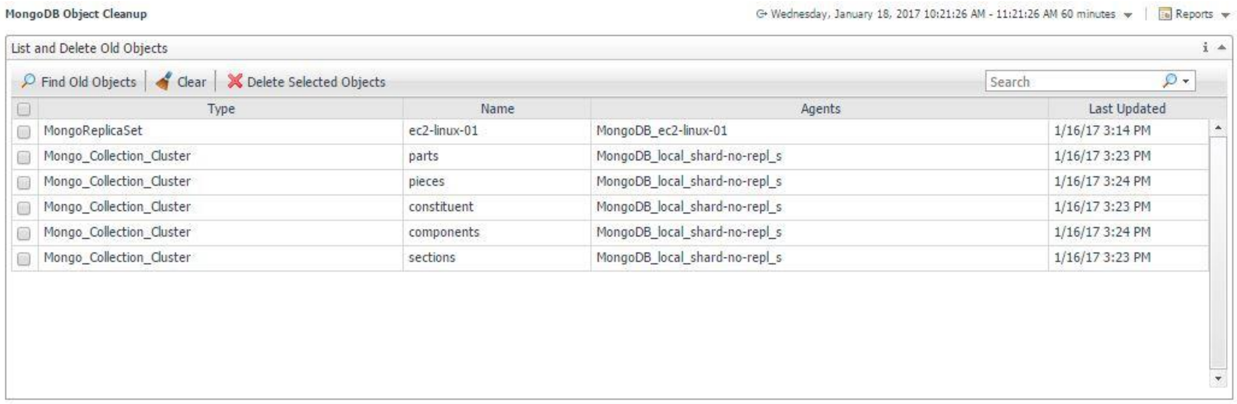
Actions
- collStats
- dbStats
- getShardVersion
- indexStats

Resource
DB:
Collection:
Cluster?
Any Resource?

Actions
- changeStream
- collStats
- dbHash
- dbStats
- find
- getShardVersion
- indexStats
- killCursors
- listCollections
- listIndexes
- planCacheRead

MongoDB Object Cleanup

This dashboard is a convenient utility for deleting MongoDB topology objects which are no longer being monitored. First, click the “Find Old Objects” button in the table header, then input the requested information and click “Find”. The Days Old field will narrow the filter to objects which have been updated in X number of days. Include Types allows you to select the topology types you want to delete. The For Agents table will include or exclude data from the selected agents depending on the option selected. Once objects have been found, they can be selected and deleted from the FMS.


MongoDB Object Cleanup
Wednesday, January 18, 2017 10:21:26 AM - 11:21:26 AM 60 minutes
Reports
List and Delete Old Objects
Find Old Objects
Clear
Delete Selected Objects
Search

□	Type	Name	Agents	Last Updated
□	MongoReplicaSet	ec2-linux-01	MongoDB_ec2-linux-01	1/16/17 3:14 PM
□	Mongo_Collection_Cluster	parts	MongoDB_local_shard-no-repl_s	1/16/17 3:23 PM
□	Mongo_Collection_Cluster	pieces	MongoDB_local_shard-no-repl_s	1/16/17 3:24 PM
□	Mongo_Collection_Cluster	constituent	MongoDB_local_shard-no-repl_s	1/16/17 3:23 PM
□	Mongo_Collection_Cluster	components	MongoDB_local_shard-no-repl_s	1/16/17 3:24 PM
□	Mongo_Collection_Cluster	sections	MongoDB_local_shard-no-repl_s	1/16/17 3:23 PM

Rules

MongoDB Alarm Email Forwarder

Forwards alarms from the MongoDB cartridge via email when they fire.

MongoDB Asserts Count

Raises an alert if any warning or user asserts are raised. While assert errors are typically uncommon, if there are non-zero values for the asserts, you should check the log file for more information. In many cases, these errors are trivial, but are worth investigating.

MongoDB Availability

Raises an alert if the monitored mongod or mongos server is unreachable two or more times in a row.

MongoDB Cleared Alarm Email Forwarder

Forwards alarms from the MongoDB cartridge via email when they clear.

MongoDB Cluster Mongos Availability

Alert when cluster is missing an active mongos.

MongoDB Collection Growth

Raises an alert if a collection has grown faster than usual. Collection size is compared to an historical average to determine if collection size growth/shrinkage is out of the ordinary.

MongoDB Connections UsedPercent

Raises an alert if the monitored instance is approaching its limit of available simultaneous connections.

MongoDB Cursors NoTimeout

Alert if there are open "no timeout" cursors.

MongoDB Database ExecutionTime Command

Alert if database command execution times are higher than usual.

MongoDB Database ExecutionTime Read

Alert if database read lock times are higher than usual.

MongoDB Database ExecutionTime Write

Alert if database write lock times are higher than usual.

MongoDB Flush Time

Raises an alert if the average amount of time the server has spent writing data to disk is high. Background flush information only appears for instances that use the MMAPv1 storage engine.

MongoDB Locks Deadlocks

Raises an alert if any deadlocks are encountered during lock acquisition.

MongoDB Locks Queue Length

Raises an alert if the combined global reader lock queue and global writer lock queue is getting long.

MongoDB Log Patterns

Alert if there are any log entries matching configured agent properties patterns.

MongoDB Memory DataFit

Alert if the total data size and index size on a server does not fit in physical memory. Must have the Infrastructure Cartridge enabled.

MongoDB Memory IndexFit

Alert if the total index size on a server does not fit in physical memory. Must have the Infrastructure Cartridge enabled.

MongoDB Memory Mapped Ratio

Raises an alert if mapped memory is too large with respect to non-mapped memory, the virtual memory used by a mongod process. With journaling enabled, non-mapped memory should be at least double the value of mapped memory. Three times larger or more may indicate a memory leak.

MongoDB Memory Page Faults

Raises an alert if the ratio of page faults to total database operations is too high.

MongoDB Profiling Operation Execution Time

Raises an alert if the average execution time for profiled operations is too high. Applicable when profiling is enabled for a given database.

MongoDB ReplicaMember PingMS

Alert if replica member pingMs is large.

MongoDB Replica Member Unreachable

Raises an alert if one or more members of a replica set are not running.

MongoDB Replica Set Not Stable

Raises an alert if a replica set member is unreachable two or more times in a row.

MongoDB Replication Buffer Ratio

Raises an alert if the replication buffer is filling up. MongoDB buffers oplog operations from the replication sync source buffer before applying oplog entries in a batch.

MongoDB Replication NoPrimary

Alert if a replica set has no primary.

MongoDB Replication Oplog Headroom

Alert if replication on a secondary is falling behind and may not have time to replicate the oldest oplog entries before they are recycled.

MongoDB Replication Oplog Lag

Alert if the replication oplog lag on a secondary server is too long.

MongoDB Server State Change

Alert if a member of a replica set changes state.

MongoDB SSL Certificate Expiration

Raises an alert if a MongoDB server using an SSL/TLS certificate is approaching its expiration date.

MongoDB WiredTiger Cache DirtyBytes

Alert if the tracked dirty bytes in the WiredTiger cache is high.

MongoDB WiredTiger Cache PagesEvictedClean

Alert if the percentage of unmodified pages evicted to the total pages currently held in the WiredTiger cache is high.

MongoDB WiredTiger ConcurrentTransaction ReadTicketsAvailable

Alert if the number of available WiredTiger concurrent transaction read tickets approaches zero.

MongoDB WiredTiger ConcurrentTransaction WriteTicketsAvailable

Alert if the number of available WiredTiger concurrent transaction write tickets approaches zero.

MongoDB WiredTiger Cursor CursorCount

Alert if there is a higher than average number of open WiredTiger cursors.

MongoDB WiredTiger Cursor SessionCount

Alert if there is a higher than average number of open WiredTiger sessions.

MongoDB WiredTiger Transaction Failures

Alert if there are any WiredTiger transaction failures due to cache overflow.

Reports

MongoDB Cluster Report Iterator

Run the MongoDB Cluster Summary report for all clusters in the monitoring environment.

MongoDB Cluster Summary

Summary of a sharded MongoDB cluster. Includes availability, version, and high-level metrics for all query servers, config servers, and shard servers.

MongoDB Executive Summary

Executive summary of a MongoDB server. Includes availability, workload, operations and alarms.

MongoDB Replica Set Summary

Summary of a MongoDB replica set. Includes availability, status, latency, and high-level metrics for all member servers.

MongoDB Server Health Check

In-depth overview of a MongoDB server. Includes availability, workload, operations and alarms.

MongoDB Service Report

Display an overview and detailed health report for each MongoDB server in a service.

MongoDB Storage Report

Shows MongoDB server storage capacity, growth rate, etc. Note: Host monitoring must be enabled in order to retrieve space remaining and days remaining until full. Days Until Full: Report on all servers that are projected to fill up before this many days. Space Remaining Percent: Report on servers with less than this percentage of remaining disk space.

MongoDB Top Collections

Top collections on a MongoDB server. Sortable by document count, size, operations, etc.

MongoDB Top Profiled Ops

Top profiled operations for a MongoDB server. Sortable by execution time, rows returned, etc.