**OTM Repository Operations Guide**

S. Livezey

December 2019

Table of Contents

[1 Introduction 3](#_Toc28694873)

[2 System Overview of the OTM Repository 3](#_Toc28694874)

[3 Routine Maintenance Tasks 4](#_Toc28694875)

[3.1 Backup of OTM Repository Data 4](#_Toc28694876)

[3.2 Upgrade of the OTM Repository Software 4](#_Toc28694877)

[4 Capacity Planning 4](#_Toc28694878)

[5 Alerts & Troubleshooting 5](#_Toc28694879)

[5.1 Server Health Monitoring 5](#_Toc28694880)

[5.2 OTM Repository Monitoring 9](#_Toc28694881)

[5.3 Indexing Manager Monitoring 11](#_Toc28694882)

[5.4 Indexing Agent Monitoring 13](#_Toc28694883)

[6 Log Files 14](#_Toc28694884)

[7 Problem Reporting 14](#_Toc28694885)

# Introduction

The OTM Repository provides an easy-to-use method for publishing and sharing OTM models among model designers and other consumers. This guide provides basic information about the support requirements for maintaining an OTM Repository in an operational production environment.

The alerts and remediation procedures described in this document assumed that the OTM Repositories are deployed and monitored using a Nagios Core server as described in the OTM Repository Installation & System Administration Guide. Many of the tasks mentioned here are described in greater detail in that document, so readers are encouraged to familiarize themselves with both guides to ensure smooth operations and support for their OTM Repository.

Because specific service-level agreements will vary from one group to another, this guide focuses on common maintenance tasks, troubleshooting, and remediation plans. The OTM Repository is licensed as open source software. It is therefore the responsibility of each individual organization to safeguard their own data and the environments where the software is deployed.

# System Overview of the OTM Repository

The OTM Repository, as described in the Installation & System Administration Guide, consists of three separate server processes:

* **Tomcat Application Server** – Used to host the OTM Repository API's and web console
* **Repository Indexing Manager** – Performs two primary functions for OTM repository indexing:
  1. Host an ActiveMQ broker used for sending and receiving indexing messages from the OTM Repository Web Service.
  2. Launch the Repository Indexing Agent process. If the agent process crashes for any reason, the Indexing Manager will relaunch it.
* **Repository Indexing Agent** – Performs the actual search indexing for the OTM repository. Indexing messages are received from the Repository Web Service (via the Indexing Manager’s Active MQ broker) when OTM libraries are created, modified, or deleted. Once the indexing is complete, notifications are sent back to the Web Service via JMS messaging.

At this time all of the above processes are designed to run locally on a single server or virtual machine. Multi-server configurations that support high availability are not currently supported for OTM Repositories.

The Nagios Core monitoring server can be configured to monitor multiple OTM Repositories. Each repository process is capable of reporting multiple application metrics via Java Management Extensions (JMX) that can be tracked and monitored remotely.

# Routine Maintenance Tasks

The following sections describe the two most common maintenance tasks for an OTM repository. While it is possible to automate one or both of these processes to occur on a regular basis, the setup of such automation is currently beyond the scope of this document.

## Backup of OTM Repository Data

As with any production system, regular backups are a recommended best-practice that prevents loss of data and the intellectual property represented by the OTM models stored in the repository. The procedures for creating an OTM Repository backup are described in the Installation & System Administration Guide. The backup process can be considered a “hot” backup that does not require any outage or downtime for users of the repository.

It should be noted that backups of an OTM Repository are actually an archive of the Subversion repository where the OTM library data is stored. Because Subversion maintains a commit history for every change that is made to the repository data, each backup archive contains a complete history of all updates ever made to the OTM models that are managed.

## Upgrade of the OTM Repository Software

Enhancements and bug fixes are continually made to the OTM-DE and Repository software. For this reason, it is strongly recommended that upgrades to the OTM Repository software are performed on a regular basis. The upgrade procedure is described in the Installation & System Administration Guide.

While the OTM Repository host itself will continue to be available during the upgrade process, many of the services that are monitored by the Nagios Core server will be down for a short time (usually less than 2-3 minutes). For this reason, it is recommended to schedule a brief downtime window for the OTM services to avoid receiving critical alerts while the upgrade is running. The procedure for scheduling downtime in Nagios Core can be found [here](https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/downtime.html).

# Capacity Planning

The most important aspect of capacity planning for an OTM Repository is the amount of available disk space. The recommended storage capacity of 30GB for the repository server is more than adequate, even for very large organizations with complex and numerous information modeling requirements.

The service monitoring described in Chapter 5 are configured to issue a warning alert when available disk usage goes above 70% (i.e. 9GB available in the recommended configuration). This should provide operations teams with plenty of time to schedule downtime and allocate additional storage for the server. This assumes that no other processes or applications are running and consuming additional storage on the server’s file system.

# Alerts & Troubleshooting

During normal setup, the Nagios Core monitoring server is configured to monitor the OTM Repository host and 21 distinct characteristics related to the application processes and the host itself. The following sections provide detailed information for each of these system properties, as well as remediation plans for each of the possible alerts that can be triggered.

## Server Health Monitoring

The following characteristics (metrics) are monitored to ensure the health of the server and operating system where the OTM Repository is deployed. It should be noted that service alerts in this section are not typically associated with an application-related issue of the OTM Repository.

|  |  |
| --- | --- |
| **METRIC:** | Server Availability (The Nagios ‘Host’) |
| **DESCRIPTION:** | Overall accessibility of the host server and operating system. |
| **CONSEQUENCE:** | Likely that OTM Repository is unavailable to OTM-DE users; web console also unavailable. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When Down |
| **CAUSE:** | A host alert is issued when the server is down or inaccessible via network ping. |
| **REMEDIATION:** | Since the repository is unavailable for use when the host is down or inaccessible, actions should be taken to correct any problems that might exist with the server or its operating environment. |

|  |  |
| --- | --- |
| **METRIC:** | VM Current Load |
| **DESCRIPTION:** | The 1-5-15 minute moving average of server workload (as reported by 'top'). |
| **CONSEQUENCE:** | Overall performance and response time degradation. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | None |
| **CAUSE:** | This is an informational metric; no alerts will be issues for high server load. |
| **REMEDIATION:** | Due to the Index Agent processing, periods of high load will be common on the OTM Repository server. If server load remains consistently high over long periods of time, additional CPU capacity should be allocated for the server. |

|  |  |
| --- | --- |
| **METRIC:** | VM Current Users |
| **DESCRIPTION:** | Number of users currently logged in to the server. |
| **CONSEQUENCE:** | By itself, no likely impact, but there is a chance of limitations on system resources and/or possible malicious intent. |
| **WARNING ALERT:** | >= 5 |
| **CRICICAL ALERT:** | >= 10 |
| **CAUSE:** | Application failure or other production incident is highly unlikely unless malicious intent exists on the part of unknown users. |
| **REMEDIATION:** | Monitor total number of users actively logged in to the server. In extreme cases, a server reboot may be required to sever connections and deactivate the logins. |

|  |  |
| --- | --- |
| **METRIC:** | VM PING |
| **DESCRIPTION:** | Indicator of server availability via network ping. |
| **CONSEQUENCE:** | Likely that OTM Repository is unavailable to OTM-DE users; web console also unavailable. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When network ping attempt fails |
| **CAUSE:** | Either server is down or is not accessible due to network connectivity issues. |
| **REMEDIATION:** | Resolve network issues or reboot VM and re-launch repository processes using the start-otm-repository.yml playbook. |

|  |  |
| --- | --- |
| **METRIC:** | VM HTTP |
| **DESCRIPTION:** | Indicator of server availability via HTTP. |
| **CONSEQUENCE:** | Likely that OTM Repository is unavailable to OTM-DE users; web console also unavailable. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When HTTP connection attempt fails |
| **CAUSE:** | Either server is down or is not accessible due to network connectivity issues. Also, possible that Tomcat server process has crashed or been shut down. |
| **REMEDIATION:** | Resolve network issues or relaunch repository processes using the start-otm-repository.yml playbook. |

|  |  |
| --- | --- |
| **METRIC:** | VM SSH |
| **DESCRIPTION:** | Indicator of server availability via SSH. |
| **CONSEQUENCE:** | Maintenance functions such as backups and upgrades of the OTM Repository will be unavailable. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When SSH connection attempt fails |
| **CAUSE:** | If not accompanied by other communication alerts, this condition is likely to be caused by a misconfiguration of the server's firewall rules. |
| **REMEDIATION:** | Resolve network issues or firewall configuration according to OS-specific procedures |

|  |  |
| --- | --- |
| **METRIC:** | VM Swap Usage |
| **DESCRIPTION:** | Indicates the % of swap space currently available (not in use) on the server. |
| **CONSEQUENCE:** | Overall performance and response time degradation. Also, possibly accompanied by repeated Indexing Agent crashes and restarts. |
| **WARNING ALERT:** | < 20% |
| **CRICICAL ALERT:** | < 10% |
| **CAUSE:** | Server is likely to be configured with insufficient physical memory. |
| **REMEDIATION:** | Allocate additional physical memory to server. |

|  |  |
| --- | --- |
| **METRIC:** | VM Total Processes |
| **DESCRIPTION:** | Total number of OS processes currently running on the server. |
| **CONSEQUENCE:** | Overall performance and response time degradation. |
| **WARNING ALERT:** | >= 250 |
| **CRICICAL ALERT:** | >= 300 |
| **CAUSE:** | Likely other processes unrelated to the OTM Repository functions running on server. |
| **REMEDIATION:** | Move unrelated processes to another server or allocated additional CPU and/or memory capacity. |

|  |  |
| --- | --- |
| **METRIC:** | VM Zombie Processes |
| **DESCRIPTION:** | Number of OS processes that have completed but still not fully exited. |
| **CONSEQUENCE:** | Overall performance and response time degradation. |
| **WARNING ALERT:** | >= 5 |
| **CRICICAL ALERT:** | >= 10 |
| **CAUSE:** | Likely other processes unrelated to the OTM Repository functions running on server. |
| **REMEDIATION:** | Move unrelated processes to another server or allocated additional CPU and/or memory capacity. |

## OTM Repository Monitoring

The following characteristics (metrics) are monitored to ensure the health of the OTM Repository Web Service deployed on the Tomcat application server.

|  |  |
| --- | --- |
| **METRIC:** | Repository Available |
| **DESCRIPTION:** | Indicates whether the OTM Repository Web Service is available |
| **CONSEQUENCE:** | OTM Repository is completely unavailable to OTM-DE users; web console also unavailable |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When the OTM Repository Web Service is unavailable |
| **CAUSE:** | System administrator action or another OS-related event. |
| **REMEDIATION:** | Restart application server using restart-otm-repository.yml playbook |

|  |  |
| --- | --- |
| **METRIC:** | Repository CPU Utilization |
| **DESCRIPTION:** | CPU utilization for the OTM Repository application server JVM |
| **CONSEQUENCE:** | Overall performance and response time degradation. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | None |
| **CAUSE:** | This is an informational metric; no alerts will be issues for high CPU utilization. |
| **REMEDIATION:** | If CPU utilization remains consistently high over long periods of time, additional CPU capacity should be allocated for the server. |

|  |  |
| --- | --- |
| **METRIC:** | Repository Memory Utilization |
| **DESCRIPTION:** | Memory utilization for the OTM Repository application server JVM |
| **CONSEQUENCE:** | Risk of Tomcat application server crash due to Out-of-Memory error, making the repository unavailable to OTM-DE users; web console also unavailable. |
| **WARNING ALERT:** | > 70% |
| **CRICICAL ALERT:** | > 90% |
| **CAUSE:** | Typically, due to a large number of concurrent processes being handled by the application server. |
| **REMEDIATION:** | Modify the startup parameters for the Tomcat application server to allocate more system memory for the JVM. May also require additional physical memory for the server. |

|  |  |
| --- | --- |
| **METRIC:** | Repository Disk Utilization |
| **DESCRIPTION:** | Disk utilization for the OTM Repository data volume |
| **CONSEQUENCE:** | Repository may be available for read operations, but all updates are very likely to fail if system disk storage runs out. |
| **WARNING ALERT:** | > 70% |
| **CRICICAL ALERT:** | > 90% |
| **CAUSE:** | Insufficient space allocated for disk volume. |
| **REMEDIATION:** | Add storage space to disk volume and restart all services using restart-otm-repository.yml playbook. |

|  |  |
| --- | --- |
| **METRIC:** | SVN Repository Available |
| **DESCRIPTION:** | Indicates whether the SVN server for the OTM repository is available |
| **CONSEQUENCE:** | Repository will be available for read operations, but all updates will fail. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When the SVN repository server process is unavailable |
| **CAUSE:** | System administrator action or another OS-related event that caused the svnserve daemon process to die |
| **REMEDIATION:** | Restart SVN server using restart-otm-repository.yml playbook |

|  |  |
| --- | --- |
| **METRIC:** | SVN Configuration Ok |
| **DESCRIPTION:** | Indicates whether SVN user credentials have been configured correctly |
| **CONSEQUENCE:** | Repository will be available for read operations, but all updates will fail. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | If the SVN configuration or user credentials for the OTM Repository are incorrect |
| **CAUSE:** | Misconfiguration of OTM Repository Web Service |
| **REMEDIATION:** | Update SVN configuration settings for OTM web service and/or SVN repository. Then restart services using restart-otm-repository.yml playbook. |

## Indexing Manager Monitoring

The following characteristics (metrics) are monitored to ensure the health of the OTM Repository’s Indexing Manager process.

|  |  |
| --- | --- |
| **METRIC:** | Index Mgr. Available |
| **DESCRIPTION:** | Indicates whether the OTM Indexing Manager is available |
| **CONSEQUENCE:** | OTM Repository loses the ability to publish indexing jobs due to the loss of the ActiveMQ broker |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When the Indexing Manager process is unavailable |
| **CAUSE:** | System administrator action or another OS-related event. The Indexing Manager is a fairly lightweight process, so this should be a rare occurrence. |
| **REMEDIATION:** | Restart indexing service using restart-otm-repository.yml playbook |

|  |  |
| --- | --- |
| **METRIC:** | Index Mgr. Memory Utilization |
| **DESCRIPTION:** | Memory utilization for the OTM Indexing Manager JVM |
| **CONSEQUENCE:** | Risk of Indexing Manager process crash due to Out-of-Memory error, causing the OTM Repository to lose its ability to publish indexing jobs due to the loss of the ActiveMQ broker. |
| **WARNING ALERT:** | > 70% |
| **CRICICAL ALERT:** | > 90% |
| **CAUSE:** | Highly unlikely for the Indexing Manager process; this would typically occur if physical memory on the server is insufficient. |
| **REMEDIATION:** | Modify the startup parameters for the Indexing Manager process to allocate more system memory for the JVM. May also require additional physical memory for the server. |

|  |  |
| --- | --- |
| **METRIC:** | Index Mgr. Last Agent Startup |
| **DESCRIPTION:** | Number of minutes since last Indexing Agent process startup |
| **CONSEQUENCE:** | None, unless repeated Indexing Agent restarts are occurring |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | None |
| **CAUSE:** | This is an informational metric that provides visibility into the amount of time that the current Indexing Agent process has been running. |
| **REMEDIATION:** | N/A |

|  |  |
| --- | --- |
| **METRIC:** | Index Mgr. Recent Agent Restarts |
| **DESCRIPTION:** | Number of Indexing Agent restarts performed by the Indexing Manager in the last 15 minutes |
| **CONSEQUENCE:** | When multiple restarts occur, indexing jobs are likely not to be processing correctly due to some kind of software bug. This is highly likely to require Level 3 (software defect) support. Libraries where indexing failures occur will appear out of date (or not at all) in search results until this failure is resolved. |
| **WARNING ALERT:** | >= 1 |
| **CRICICAL ALERT:** | >= 4 |
| **CAUSE:** | Likely causes are indexing of large libraries with many cross-library dependencies, since such models require large amounts of memory to index. If not related to model/library size, the most likely cause is a software defect. |
| **REMEDIATION:** | First action should be to reconfigure the indexing launch scripts to allocate more JVM memory. Then restart indexing service using restart-otm-repository.yml playbook. If that does not work, it is likely that a software defect exists for which a fix will be required. |

## Indexing Agent Monitoring

The following characteristics (metrics) are monitored to ensure the health of the OTM Repository’s Indexing Agent process.

|  |  |
| --- | --- |
| **METRIC:** | Index Agent Available |
| **DESCRIPTION:** | Indicates whether the OTM Indexing Agent is available |
| **CONSEQUENCE:** | Indexing Manager will automatically restart the Agent process. No impact if restarts are infrequent (i.e. limited to a single occurrence over a relatively long period of time). |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | When the Indexing Agent process is unavailable |
| **CAUSE:** | Likely causes are indexing of large libraries with many cross-library dependencies, since such models require large amounts of memory to index. If not related to model/library size, the most likely cause is a software defect. |
| **REMEDIATION:** | No action required if Indexing Agent restarts are infrequent. |

|  |  |
| --- | --- |
| **METRIC:** | Index Agent Memory Utilization |
| **DESCRIPTION:** | Memory utilization for the OTM Indexing Agent JVM |
| **CONSEQUENCE:** | Risk of Indexing Manager process crash due to Out-of-Memory error, causing the Indexing Manager to automatically relaunch the Agent process. This is only a concern if repeated crashes of the Indexing Agent process occur in a short period of time. |
| **WARNING ALERT:** | > 70% |
| **CRICICAL ALERT:** | > 90% |
| **CAUSE:** | May sometimes happen during indexing of large OTM libraries with numerous complex dependencies. |
| **REMEDIATION:** | At this time, there is little that can be done to resolve this issue since the Indexing Agent process is launched by the Indexing Manager. Future enhancements to the Indexing Manager may provide a means of controlling the memory allocation for the Agent process. |

|  |  |
| --- | --- |
| **METRIC:** | Index Agent Item Count |
| **DESCRIPTION:** | Number of OTM libraries indexed since the last service check (default configured for every 10 minutes) |
| **CONSEQUENCE:** | None, unless this metric always reads zero after updates have been committed to the repository. Such a level of inactivity is a sign of functional defects in the Indexing Agent that should be reported as a software defect. |
| **WARNING ALERT:** | None |
| **CRICICAL ALERT:** | None |
| **CAUSE:** | This is an informational metric that provides visibility into the level of activity being performed by the Indexing Agent process. |
| **REMEDIATION:** | N/A |

# Log Files

The following log files can provide useful information regarding the use and operations of the OTM Repository:[[1]](#footnote-1)

|  |  |
| --- | --- |
| Log File Location | Description |
| /opt/otm-repository/tomcat/logs/catalina.out | Application log for the OTM Repository Web Service. |
| /opt/otm-repository/  tomcat/logs/localhost\_access\_log.<date>.txt | Access log for the Tomcat server that provides basic information about the requests and transactions that have been received. |
| /opt/otm-repository/  ota2-indexing-service/logs/indexing-manager.log | Application log for the Indexing Manager process. |
| /opt/otm-repository/  ota2-indexing-service/logs/indexing-agent.log | Application log for the Indexing Agent process. |

# Problem Reporting

Functional defects and other operational stability issues with the OTM Repository can be reported [here](https://github.com/OpenTravel/OTM-DE-Compiler/issues) on the OTM-DE-Compiler GitHub project. Alternately, problems can be reported by contacting the OpenTravel Alliance team via email at [info@opentravel.org](mailto:info@opentravel.org).

1. The log file locations provided assume that the default directory locations were used for installation and setup of the OTM Repository. [↑](#footnote-ref-1)