Oracle Metadata Management (OMM) Solutions

Oracle Metadata Management for Oracle Business Intelligence (OMM4OBI)
Oracle Enterprise Metadata Management (OEMM)

OMM Metadata Management

with Meta Integration® Metadata Management (MIMM)

README for Release Notes, Installation & Setup

Table of Contents

1. Overview

The Oracle Metadata Management (OMM) solutions include two products:

- the Oracle Metadata Management for Oracle Business Intelligence (OMM4OBI)
- and the Oracle Enterprise Metadata Management (OEMM)

Oracle Metadata Management for Oracle Business Intelligence is a software package for metadata management of Oracle environments. Oracle Metadata Management for Oracle Business Intelligence includes the following metadata management features:

- Metadata Harvesting from Oracle technologies
- Metadata Configuration and Stitching
- Metadata Browsing, Search and Reporting
- Metadata Collaboration (external URL, tagging, comments and review)
- Data Flow Lineage & Impact Analysis
- Metadata Explorer (simplified metadata user interface for business users)

Oracle Enterprise Metadata Management is a software package for metadata management of multi-vendor environments and support for data governance. Oracle Enterprise Metadata Management includes all features of Oracle Metadata Management for Oracle Business Intelligence with the following extra metadata management features:

- Metadata Harvesting from multi-vendor technologies
- Metadata Version and Configuration Management (change management)
- Data Model Diagram Visualizer and Navigator
- Business Glossary for Data Governance
- Semantic Lineage & Impact Analysis
- Semantic Mapping Editor
- Data Flow Mapping Specifications Editor

The above Oracle Metadata Management Solutions are implemented by Meta Integration® Metadata Management (MIMM) Web Application Server, based on a Meta Integration® Repository (MIR) database server, and the Meta Integration® Model Bridge (MIMB) metadata harvesting components.

2. Copyright Notice

The following Oracle Metadata Management (OMM) products:

- Oracle Metadata Management for Oracle Business Intelligence (OMM4OBI)
- Oracle Enterprise Metadata Management (OEMM)

are licensed under the following Oracle copyright:

These Oracle Metadata Management products are based (re-branded OEM) on the following Meta Integration products:

- Meta Integration® Metadata Management (MIMM)
- Meta Integration® Model Bridge (MIMB)
- Meta Integration® Repository (MIR)

which are licensed under the following Meta Integration copyright:

3. Release Changes

OMM 12.2.1.1 based on MIMM OEM 9.0.2 (06/10/2016)

- METADATA MANAGER:
  - Major improvements in MODEL VERSION / CHANGE MANAGEMENT preventing the creation of unnecessary new versions of models if the source metadata (e.g. a database, or data model) has not changed since the last automatically scheduled metadata harvesting. This is new feature is achieved by taking advantage of the MIMB's new capabilities to compare the metadata of a newly imported model with a previously imported one in order to detect any change. The major benefit of this new feature is to dramatically reduce the disk space in the repository by automatically deleting unnecessary versions.
  - New CONFIGURATION / VERSION CHANGE MANAGEMENT capabilities offering a comparator of versions of configurations.
- METADATA EXPLORER:
  - Major redesign of the Metadata Explorer UI on both the look & feel and actual capabilities.
  - New METADATA SEARCH/BROWSE with FILTERING AND REPORTING capabilities offering:
    - New Search (or Browse) with Filtering capabilities on any attributes/properties
    - New choice of result display as a classic Google like "LIST", or a new powerful "GRID" offering a spreadsheet like display of multiple attributes/properties at once. Such attributes/properties can be individually selected, reordered, and sorted, basically offering a full metadata reporting solution. The results can of course be downloaded as CSV/Excel files.
  - New METADATA EDITING Capabilities at many levels including:
    - Numerous new fast and easy "in place" editing to Rename objects, Edit Descriptions, and more.
    - The new Search/Browse GRID display also offers efficient editing with:
      - TABULAR EDITING of multiple objects at once such as Business Glossary Terms, Data Models Tables, or Table Columns
      - BULK CHANGE of multiple objects at once, where a search can return multiple objects (that can then be selectively subsetted) for which changes can be performed at once (e.g. Change the Security Threat Level to Orange to a set of tables at once)
- METADATA AUTHORING TOOLS:
  - Common Features:
    - Major improvements in CUSTOM ATTRIBUTES (also known as User Defined Properties) to objects of the Business Glossaries and Data Models
      - Custom Attributes are now common to the MM repository (e.g. Security Threat Level = [Red, Orange, Yellow, Blue, Green] and are therefore shared between Business Glossaries, Physical Data Models, etc. Therefore, having a centralized place for maintenance (e.g. adding a new value: purple)
      - Custom Attributes can now have a default value (e.g. default value is green)
      - Custom Attributes now have a much wider scope to be applied at any level from a high level repository object (e.g. harvested Model, Data Mapping, Directory) to fine grain model objects (e.g. A Business Glossary / Term, and/or a Physical Data Model / Column)
      - Custom Attributes now have a security group associated to them (e.g. the Security Threat Level custom attribute may only be set by a custom Security Approved group)

        - New AUDIT TRAIL for any changes in objects of Business Glossaries and Data Models, including who changed a given attribute and when
    - BUSINESS GLOSSARY:
        - New Business Glossary editing capabilities are now available to the business users under the Metadata Explorer UI. (includes tablet friendly in place editing, as well as HTML formatting of descriptions, etc.)
    - DATA MODELING:
        - PHYSICAL DATA MODEL (data documenter for existing data stores databases, data warehouse, data lake):
            - NEW feature with this version of OMM
    - DATA MAPPING:
        - DATA MAPPING SPECIFICATIONS (Data Flow Mapper)
            - Minor improvements and bug fixes
- ARCHITECTURE & TECHNOLOGY:
    - Major database performance improvements
    - Updated MIMM Web Services which are now based on RESTful API technology (i.e. therefore removing any security vulnerabilities of the older Axis technology)

OMM 12.2.1.0 based on MIMM OEM 9.0.1 (12/15/2015)

- ARCHITECTURE & TECHNOLOGY:
    - 100% Java delivery and installation allowing support for Windows as well as a variation Linux/Unix deployments
        - The Metadata Management Server (OMM) can now be installed on Unix/Linux variations.
        - The Metadata Harvesting Agent (MIMB bridges) can be installed:
            - Locally (co-located with a OMM server on Linux) to run 100% java based bridges including JDBC database bridges (Oracle, Teradata, DB2, SQL Server, etc.), big data bridges (Hadoop Hive, HCatalog), and other popular bridges such as CA ERwin xml, Informatica PowerCenter xml, Tableau BI, etc.
            - Remotely on a Windows machine for C++ based bridges and COM API based bridges requiring software SDK running on Windows such as CA ERwin native (.erwin) files, and many BI tools like SAP BO Universe, Microstrategy, QlikView, etc.
    - 100% HTML 5 (no more Flash) and tablet friendly look & feel, allowing to run on Mac, Tablets, and more
    - Metadata Explorer Customization to offer a better experience to targeted business users of the customer company:
        - Customize headers, company logos, menus, search categories, home page (with search widgets), automatic opening of reports (BI portal experience, BI report documentation (adding BG terms)
    - Improved Metadata Explorer Search Performance
    - Improved Data Integration (DI/ETL/ELT Import Bridge) Harvesting Performance
        - Now offering detailed DI data flow lineage analysis on demand only (in real time), instead of pre-calculating even if unused

OMM 12.1.3.0.2 based on MIMM OEM 8.0.3 (05/19/2015)

- DATA MODEL DIAGRAM VISUALIZER:
    - Major improvements with the HTML5 redesign including better scalability, performance and overall layout quality
    - New interactive search
    - New diagram auto layout
    - New dynamic layout of a diagram subset starting from an entity with all related entities with one or two levels of relationships (very useful for large diagrams)
- SEMANTIC MAPPER:
    - New support for BI Report metadata (such as a page of a workbook, a table or pie chart on page, or just the axis of a graph), allowing to document precise items within a given report by associating (semantic links) business glossary terms to them.
    - New support for models within a multi-model server source, allowing to provide documentation (or glossary) at the high level of a given data model (or BI report) within a multi model server, such as:
        - a Data Modeling (DM) repository server with many data models inside (e.g. CA ERwin Mart)
        - a Business Intelligence (BI) content server with many BI designs and BI reports inside (e.g. SAP BusinessObjects)
- BUSINESS GLOSSARY:
    - New customizable role driven workflow support (can be turned off) and security enforcement
- METADATA EXPLORER UI:
    - New integrated presentation of Business Glossary terms related to any data store or BI report objects, including the ability to add/remove BG terms documenting a data store or BI report
    - Improved search support for auto complete and objects
- ADMINISTRATION:
    - New group based security model (as side effect of the new role based Business Glossary workflow)

- ARCHITECTURE & TECHNOLOGY:
  - New support for HTML5 only devices like iPad and other tablets (Flash will no longer be needed)
    for graphically tracing any data flow or semantic lineage (Lineage Analyzer),
    and for visualizing data models (Diagram Visualizer).
  - Java 8 (compiled with backward compatibility with Java 7) compliance (Java 6 is no longer supported)

OMM 12.1.3.0.1 based on MIMM OEM 8.0.1 (12/02/2014)

- FEATURES:
  - New "Show Related Reports" (e.g. from a Glossary Term)
- METADATA MANAGER UI:
  - New Metadata Manager look & feel (to match the Metadata Explorer)
  - New Business Glossary batch editing
- METADATA EXPLORER UI:
  - New customizable action menus per repository object type (e.g. open BI report with BI tool by default)
  - New dedicated web pages for tracing data lineage & impact, and semantic definition & usage
  - New access to the Configuration's Enterprise Architecture Diagram

OMM 12.1.3.0.0 based on MIMM OEM 8.0.0 (10/01/2014)

- Initial Release

4. System requirements

4.1 Important preliminary disclaimer notice on all requirements

The following requirements only define the minimal requirements to run the application server with reasonable performance based on the provided tutorial, or small business use cases. The actual requirements for enterprise wide use cases based on larger models and configurations do require significantly greater resources to obtain acceptable performance.

The following requirements are based on:

- actual physical hardware (no virtual environment),
- minimal to no network overhead (assuming both the database and Application servers to be locally installed),
- vendor's default install of the current version of their software (with all current service or fix packs),
- no other applications sharing such hardware (starting from a clean machine),

Any other hardware/software configurations are acceptable as long as they provide the same (or better) results on the provided performance benchmark. In such case, if any problem is discovered (e.g. scalability or performance issues), then the customer must be able to reproduce the issue using an environment that conforms to the minimum performance requirements as defined herein.

Potential known issues include (but are not limited to) the following:

- actual usable hardware performance on virtual environments (e.g VMWare configuration and licenses)
- network overhead on remote servers (e.g. bandwidth, proxy, VPN issues, VMWare inter OS network limitations without a proper license, etc.)
- shared resources with competing applications on the same OS, or between OS on a virtual environment,
- licensing limitations (e.g. most database server licenses limit the number of usable core/CPU)
- vendor software known limitations and requirements (e.g. Oracle on VMWare vs Oracle VM)

4.2 Web Client requirements

Users only need an internet browser:

- Google Chrome v30 or newer
- Microsoft Internet Explorer (IE) v10 or newer. Note that older versions of IE (especially running on old versions of Windows like XP) are not officially supported as users may encounter some minor layout issues for this modern web 2.0 application (all issues have been resolved by Microsoft in more recent versions of Windows and IE).
- Mozilla Firefox v30 or newer
- Apple Safari v6 or newer

4.3 Application Server Requirements

Hardware Minimum Requirements (based on physical hardware performance, not a virtual environment):

- 2 GHZ or higher quad core processor
- 8 GB RAM
- 10 GB of disk space (all storage is primarily in the database server)

Operating System Requirements:

- Microsoft supported Windows 64 bit versions (including Windows 2008 Server, Windows 2012 Server, Windows 7, Windows 8.x, and Windows 10).

    - Ensure that installer is executed with full Administrator privilege
    - Ensure that Microsoft .NET Framework 3.5 or higher is installed
    - Ensure that all current Microsoft Windows critical updates have been applied
- Most popular Linux/Unix 64 bit Operation System Versions (such as Oracle Solaris, Redhat or Mac OS).

Application Server Engine Requirements:

- Apache Tomcat 7 - 64 bit (bundled)
- Other Application Servers (such as IBM WebSphere or Oracle WebLogic) require manual install/setup, and are therefore not supported by this version.

Java Runtime Environment (JRE):

- Oracle JRE 8 - 64 bit (bundled and recommended)
- Other Java Runtime Environment (JRE) (such as IBM Java) require manual install/setup, and are therefore not supported by this version.

4.4 Database Server Requirements

Hardware Minimum Requirements (based on physical hardware performance, not a virtual environment):

- 2 GHZ or higher quad core processor
- 8 GB RAM
- 20 GB of disk space (or more as needed for the data)

Database Administrator privileges are required to install/uninstall the database.

The OMM Database Server can reuse your existing Oracle database server:

- **Oracle 10g R2 to 12c - 64 bit** (recommended for large enterprise, default supported version)
    - The character set of the database must be AL32UTF8 (UTF8); because the Oracle InterMedia Search can only index columns of type VARCHAR or CLOB (not the national variants NVARCHAR and NCLOB respectively)
    - The CTXSYS user must be installed: the installation script can be found in <ORACLE_HOME>/ctx/admin/catctx.sql
    - In order to find out what exact Oracle edition/version is actually installed:
      ```
      sqlplus.exe SYS@<DB-NAME> as SYSDBA
         select banner from v$version where BANNER like '%Edition%';
      ```
    - In order to find out how much memory is actually available to the Oracle database, it is important to first understand how Oracle's memory is configuration and used:
        - The actual available System Global Area (SGA) memory can be found using:
          ```
          sqlplus.exe SYS@<DB-NAME> as SYSDBA
             show sga;
             select * from v$sga;
             select * from v$sgainfo;
          ```
        - The actual available Program Global Area (PGA) memory can be found using:
          ```
          sqlplus.exe SYS@<DB-NAME> as SYSDBA
             select * from v$pgastat;
          ```
    - In order to find out how much processing CPU/Cores is actually available to the Oracle database, query the table v$parameter for the value of cpu_count, or query the table v$license as follows:
      ```
      sqlplus.exe SYS@<DB-NAME> as SYSDBA
         select * from v$license;
      ```

In general, one must ALWAYS install the latest service packs for a given database version BEFORE creating the OMM database. E.g., for Oracle 11.2 one is required to apply the patches to upgrade to 11.2.0.3, or whatever is the latest patch level at the time. In addition, Oracle 11.2.0.4 must have patch 17501296 applied.

Virtual Memory: For a Windows based database server, be sure to either:

- set the page file size to be managed automatically by OS
- or it should be at least 3 times the memory or RAM size for the machine.

Thus, you must have more than that much free disk space (at least 3 time the amount of memory or RAM) on the drive where the page file is defined to reside.

5. Metadata Management (OMM) Database Server Setup

The OMM Application Server requires the connection to an existing Database server for metadata storage (metadata repository)

The following database setup scripts and instructions assume the following by default:

```
   Database Name: MM
   Database User: MM
   Database Password: = MM123!
```

The database name and user name can be changed, and the password should of course be different.

After the product is fully installed and web connectivity has been made, one may connect to a different database by way of the web based user interface at Tools -> Administration -> Database.

5.1 Database on Oracle

Create a user MM and a database MM with the following privileges:

```
sqlplus.exe SYS@<DB-NAME> as SYSDBA

   -- Delete previous user and database if needed
   -- DROP USER MM CASCADE;

   CREATE USER MM IDENTIFIED BY MM123!;

   GRANT CONNECT TO MM;
   GRANT CTXAPP TO MM;

   GRANT CREATE TABLE TO MM;
   GRANT CREATE VIEW TO MM;
   GRANT CREATE SEQUENCE TO MM;
   GRANT CREATE TRIGGER TO MM;
   GRANT CREATE PROCEDURE TO MM;
   GRANT CREATE TYPE TO MM;

   GRANT EXECUTE ON CTXSYS.CTX_DDL TO MM;
   GRANT EXECUTE ON DBMS_LOB TO MM;
   GRANT EXECUTE ON SYS.DBMS_LOCK TO MM;

   -- If you get the error "Database exception occurred: ORA-01950: no privileges on tablespace 'USERS'"
   -- ALTER USER MM QUOTA UNLIMITED ON USERS;
```

Advanced Oracle 12 DB Administrator may also optimize the KEEP buffer pool. For more details, please refer to:

```
   %OMM_HOME%\tomcat\conf\localhost\MM.xml
```

6. Metadata Management (OMM) Application Server Setup

6.1 Application Server Installation and Configuration

The OMM Application Server is installed as follows:

- On **Windows** operating systems, use `unzip` to extract the software package (.zip) in the directory of your choice. You should avoid using the "Program Files" directories of Windows 7, 8.x and 10 as they have are now controlled by Windows with special access rights. Depending, on your software installation directory, you may need "Administrator" privileges.
- On **Linux** operating systems, use `tar -xjvf` to extract the software package (.tbz2) in the directory of your choice. Depending, on your software installation directory, you may need "root" privileges.

If your are using an existing database and do not wish to customize the application server (e.g. memory allocation, Windows services), then you can skip this step and go directly to the section on Application Server Execution and Initialization

Otherwise, go to the software home directory and "run As Administrator" the `Setup` utility (.bat on Windows or .sh on Linux). This setup utility will allow you to setup the configuration parameters defined below through a user friendly application. After any change on any panel (tab) below, remember to press the **Configure** button in order to perform the configuration changes. A dialog box will be issued to confirm success or failure (with error messages). Alternatively, this setup utility also works at the Windows command line or Linux shell, use the `-help` the options.

- **Product Edition** tab:
  - **Oracle Enterprise Metadata Management (OEMM)**
  - **Oracle Metadata Management for Oracle Business Intelligence (OMM4OBI)**
- **Application Server** tab:
  - **Enable Windows Service**
    This will create the "OMM Application Server" Windows Service, set it for automatic start, and actually start it. Unchecking that box will delete the "OMM Application Server" Windows Service, which is a good idea before uninstalling the OMM software.
  - **Metadata Harvesting Server Only**
    This allows to setup this application server as a metadata harvesting server only, rather than a full metadata management server. This is very useful in architecture deployments where the metadata management server is deployed on Linux, but needs to access remote metadata harvesting servers (agents) on Windows machine where DM/DI/BI client tools are Windows only (e.g. COM based SDK).
  - **Metadata Harvesting Browse Path**
    This controls the access to the file system for metadata harvesting. The default value is set to '*' which means any Windows drive (C: and any mounted remote drive R:) or any directory from root on Linux. It is strongly recommended to limit the access to a common shared data location, and avoid system area.
  - **Data Directory**
    This is the location of all data files, including log files as well as the metadata harvesting caching. The data directory is located by default in the 'data' subdirectory of the application server home directory. It is recommended to separate the program data from the program files, this allows you to provide a new location for the data in a separate area (with regular backups if possible). Note that changing to a new location will not move the existing data from the previous location. Either the new location already had the data (from a previous install), or new data will be created.
  - **Max Memory**
    This defines the maximum memory used by Java (JRE) on the OMM Application Server (Apache Tomcat). This is unrelated to the maximum memory used by java on bridges for Metadata Harvesting which is separately set by default with the M_JAVA_OPTIONS variable in %OMM_HOME%\conf\conf.properties, and can be overridden within the Miscellaneous parameter of memory intensive import bridges (e.g. JDBC).

- **Port Number**
  This set to a custom port number by default to avoid conflicts with other web application servers. However, this can be set back to 80 to avoid having to specify any port number in the URL.
- **SSL**
  This enables Secure Socket Layer (SSL) communication for web access (HTTPS). In order to support HTTPS, the OMM Tomcat service must be configured to work with HTTPS for encryption of passwords and other content exchanged between the web client and the OMM Application Server. In this case, you will need a certificate for the HTTPS protocol to work. Note: the OMM software does not perform any error handling for validating a certificate associated with the OMM Application Server, and most web browsers will report an error if the certificate is not provided by a valid certificate authority. Thus, your certificate should be a trusted certificate provided to you by a valid Certificate Authority.
  - **Certificate file**
    Mandatory
  - **Root Certificate file**
    Optional (only required if the above certificate file was generated by an external company as a certificate authority)
  - **Key file**
    Mandatory
  - **SSL Key Password**
    Optional (only required if the above key file is password protected)

6.2 Application Server Upgrade

6.2.1 Understanding the Data Locations

Most application data is obviously located and your database server, you are responsible for regular backup of such database. However, it is also important to understand that the software installation directory (known as `%OMM_HOME%` in this document) also contains some critical application data and application setup customizations that have to be taken into account in your backup or upgrade process, including:

- `%OMM_HOME%\data`
  which contains other application data such has the metadata harvesting cache (critical for incremental harvesting, and metadata export), the application server cache (Tomcat), the log files (metadata harvesting with MIMB and Tomcat), and other temp files. Remember that the actual location of this OMM Application Server data directory can be configured with the Setup utility (in the "Application Server" tab).
- `%OMM_HOME%\conf`
  with the `conf.properties` file containing most customizations defined with the Setup utility (in the "Application Server" tab), and the `\ressources` directory containing any User Interface Customizations.
- `%OMM_HOME%\tomcat\conf`
  with the `tomcat.properties` file containing the tomcat port and memory customizations defined the Setup utility in the "Application Server" tab, and the `keystore` file containing the tomcat SSL certificates defined with the Setup utility (in the "Application Server" tab).
- `%OMM_HOME%\jre\lib\security`
  which also contains some SSL customizations defined with the Setup utility (in the "Application Server" tab). It is recommended to not reuse such directory, but rather reinstall the SSL keys with the Setup utility.

6.2.2 Upgrade Process

We recommend the following upgrade process:

- Stop the OMM Application Server (possibly using the Windows services).
- Backup your previous installation by renaming the `%OMM_HOME%` directory as `%OMM_HOME%.old`
  and then install the new software package at the exact same previous location: `%OMM_HOME%`
- Restore your data and customization/setup by copying the appropriate files and directories (as previously explained) from `%OMM_HOME%.old` to `%OMM_HOME%`, including at least `\data` and `\conf\conf.properties` but possibly more as used and customized such as `\conf\ressources`, or `\tomcat\conf`.
- Finally, restart the OMM Application Server (possibly using the Windows services) which may prompt you for a database upgrade of the OMM tables.

6.3 Application Server Execution and Initialization

The easiest way to start the OMM Application Server is to go to the software home directory and use the `RestartApplicationServer` utility (.bat on Windows or .sh on Linux).

- On **Windows** operating systems, you can alternatively use the Windows Services to control the OMM Application Server by using the `RestartApplicationService.bat` utility instead. This utility will create the Windows Service for the OMM Application Server, if it was not already created by previous execution of this utility or the `Setup.bat` utility. At this point, you can simply use the Windows Services to start, stop or restart the OMM Application Server automatically.
  When running the OMM Application Server as a Windows Service, it is important to configure the user running such service in order to have full access rights to the needed files and applications. For example, the MIMB bridges involved in the metadata harvesting may need to invoke the SDK of third party software such as the COM based API of CA ERwin, or SAP BusinessObjects Universe Designer.
  In order to set such access rights, go to the services manager of Windows, right-click on the OMM Application Server service. Then, go to the "Log On" tab to define an account by name under which the service will run.
- On **Linux** operating systems, administrators can use the system daemon directories (e.g. `/etc/init.d/` or `/etc/systemd/`) to control the OMM Application Server (either using the `RestartApplicationServer.sh` utility or directly controlling the tomcat server in the home directory).

The final initialization steps of the setup are performed over the web browser as follows:

1. **Connection**
   Connecting to the server on Windows can be simply achieved by opening the `Metadata Management` link in the home directory. In all cases, you can connect to the server using your internet browser to open by default: http://localhost:11580/MM. Note that the default port of this URL number may have been changed by the `Setup` utility in the section Server Configuration..

2. **Database**
   Define the connection to the previously created database (in the above steps), by providing the database type, user, password, and URL (JDBC connection). Press **`Test Connection`** button to verify proper database connectivity. Finally, when the pressing the **`Save`** button, the OMM Application Server will create all the necessary tables in the database.

3. **Login**
   Login as "Administrator" with password "Administrator". Note that you should change that password later in the application by going to: **`Tools -> Administration -> Users`**)

6.4 Custom integration with authentication environments

OMM is able to support three authentication methods:

1. Native Authentication, where the password is managed by the software and stored within the database.

2. LDAP Authentication, where the software does not manage or store the LDAP passwords at all. Instead, it is simply passed it through to LDAP in order to authenticate.

3. External Authentication such as Single Sign On (SSO), where the software does not perform any authentication, and leaves that responsibility to a local single sign on service managed by the customer.

In Tools->Administration->Users one may specify either:

1. Mixed Native and LDAP authentication where users may be authenticated either as native users or LDAP users

2. External authentication where the system does not perform any authentication, leaving it up to a local Single Sign On environment.

6.4.1 Native Authentication Configuration Issues

There are no specific configuration steps for Native Authentication.

6.4.2 LDAP Authentication Configuration Issues

There are no special server configuration issues for LDAP Authentication. LDAP connectivity configuration is documented in the online help.

6.4.3 Windows Authentication Issues

It is also possible to enable OMM and Tomcat to obtains authentication for users from Windows authentication via the browser (client). This way, users will automatically be authenticated if they are running from a Windows session.
To do so, one must install a third party product named Waffle (Windows Authentication Functional Framework) as an addon (see here).

1. Go to Tools->Administration->Users->LDAP and ensure that all LDAP settings are cleared

2. Unzip the Waffle zip.

3. Copy all the jar files from it to **`%OMM_HOME%\tomcat\lib`**

4. Open **`%OMM_HOME%\tomcat\conf\web.xml`**. Search for "Windows authentication support". Uncomment the block following that.

5. Restart OMM.

6. You should have windows authentication enabled now. Any valid windows user will be logged in as guest by default as long as licensing allows it. If you need to get an administrator interface, you can access: http://host:port/Admin

7. Provide connection information for the database you created above.

Note: Automatic Windows authentication will not allow one to use the browser refresh (f5) with IE 8.x when used as the client browser. Refresh will force a re-authentication on IE 8.x browsers and will not be automatically authenticated. If this occurs, the user must close all instances of the browser and start again.

To avoid this issue, one must use IE 9.x or later or another approved browser (see System requirements)

In addition, for Internet Explorer and Firefox, you must configure the browser at each client to support automatic Windows authentication. Please refer to the Waffle web site here.

6.5 Custom integration for Secure Socket Layer (SSL) communication

Important Disclaimer: SSL is primarily used for HTTPS secure communications from the web browser clients to the OMM Server itself. Such common HTTPS setup can be fully achieved with the **`Setup`** utility as explained in Server Installation and Configuration. The following steps are provided for illustration purpose only (manual steps), describing what the **`Setup`** utility already performs automatically. THEREFORE, YOU DO NOT HAVE TO PERFORM THESE STEPS BELOW.

If you want to manually install a your own certificate, you must:

1. Change the referenced (in server.xml) connector entry parameters (keystoreFile and keystorePass) to point to the correct keystore file and password.

2. Import that certificate into the JRE that is being used by this tomcat. The default JRE is located under:
   ```
   %OMM_HOME%/jre.
   ```

3. Use the following commands:
   ```
   cd %OMM_HOME%/jre/lib/security
   move jssecacers jssecacers.old
   %OMM_HOME%/jre/bin/keytool -importkeystore -srckeystore {your_keystore} -keystore jssecacerts
   %OMM_HOME%/RestartApplicationServices.bat
   ```

After the configuration, use the default URL to Access OMM: https://localhost:11580/MM

Or use the ports specified in the server.xml file. For example:

```
<Connector port="11580" maxThreads="200"
           scheme="https" secure="true" SSLEnabled="true"
           keystoreFile="conf\keystore" keystorePass="changeit"
```

```
            clientAuth="false" sslProtocol="TLS" />
```

6.5.1 Configuring OMM to securely connect via HTTPS to another OMM server for Metadata Harvesting

Important Disclaimer: the following steps are needed ONLY IF you use a self signed certificate for SSL (WHICH IS NOT RECOMENDED), AND ONLY in the case of configuring OMM to securely connect via HTTPS to another OMM server for Metadata Harvesting. Only in such exceptional use case, then the following additional steps have to be performed

In order to support HTTPS from a OMM Server acting as the "Metadata Manager" to a OMM Server acting as "Metadata Harvesting" Agent, the Administrator needs to import the trusted certificate that the OMM "Harvesting Agent" Server is using into the JRE that the OMM "Metadata Manager" server is using. The following page describes the process: http://docs.oracle.com/javase/tutorial/security/toolsign/rstep2.html.

The command looks like the following:

```
cd %OMM_HOME%\jre\lib\security
..\..\bin\keytool.exe -import -alias john -file YourOwnCertificate.cer -keystore jssecacerts
```

6.5.2 Configuring OMM to securely connect via LDAPS to the Enterprise Directory

In LDAP Authentication, the user password is not managed by the software and is simply passed through to the LDAP system.

Note: this password is not encrypted when communicated between the client and the server. Thus, in order to ensure encryption you may wish to specify HTTPS protocol communication, as above.

Note: this password is also not encrypted when communicated between the server and LDAP. Thus, in order to ensure encryption you may wish to also specify LDAPS protocol communication and thus use SSL to encrypt.

In order to support LDAPS, the OMM Tomcat service does not itself need to be configured to work with LDAPS for encryption of passwords. However, to enable secure SSL communication between OMM and LDAP servers the Administrator needs to import the trusted certificate that the LDAP server is using into the JRE that the OMM Application server is using. The following page describes the process: http://docs.oracle.com/javase/tutorial/security/toolsign/rstep2.html.

The command looks like the following:

```
cd %OMM_HOME%\jre\lib\security
..\..\bin\keytool.exe -import -alias john -file YourOwnCertificate.cer -keystore jssecacerts
```

This is an entirely different certificate from the one used by the HTTPS protocol.

6.5.3 SSL Security Vulnerabilities

Poodle is a "Man In The Middle" (MITM) vulnerability which needs to be primarily fixed server side. An attacker can trick the server into downgrading the encryption protocol used to communicate. The servers should be configured to disallow TLS fallback, or to disable SSLv3 as a valid protocol.

If Tomcat has been configured with SSL support, the customer should add the following to the connector description in the %OMM_HOME%\tomcat\conf\server.xml
   sslEnabledProtocols="TLSv1.2,TLSv1.1,TLSv1"

7. Model Bridge (MIMB) Metadata Harvesting Setup

The Metadata Integration or Metadata Harvesting from third party databases, data modeling, data integration or business intelligence tools is performed by the integrated Meta Integration® Model Bridge (MIMB) software. By default, the installer software deploys and configures both OMM and MIMB on the same Windows machine, where the OMM Application Server accesses the MIMB Web Services locally. MIMB can also be installed and configured as a remote MIMB Agent on another machine: for example on a Windows machine where a Windows only third party software is needed by an MIMB bridge (such as SAP BuinessObjects universe).

Essential customizations (e.g. directories, memory) of the MIMB Application Server can be performed in the following configuration file:
   `%OMM_HOME%\conf\conf.properties` Recommended customizations include:

- M_BROWSE_PATH to browse local and mapped network drive.

  All metadata harvesting file and directory parameter references are relative to the server. The reason is that the server must have access to these resources anytime another event (e.g., scheduled harvest) is to occur. When harvesting a model, then, the UI presents a set of paths that may be browsed in order to select these files and directories. Setting the M_BROWSE_PATH parameter allows one to define which drives and network paths will be available in the UI. One may update the M_BROWSE_PATH using the UI (on the application server) presented by the setup.bat (or setup.sh on Linux) command (see also Application Server Execution and Initialization), or by editing the %MIMM_HOME%\conf\conf.properties file directly.

  On installation, the set includes all directly attached drives., which is specified by an asterisk "*" (M_BROWSE_PATH=*).

  Note for Windows based application servers: When running as a service, the drive names (mapped) and paths may not be the same as what a user sees when logged in, and thus the "*" value will not be see all drives you might expect when selecting drives using the UI. Instead, one must explicily list all the drives and network paths that one wants to be available to all users in the UI. Also, it is not sufficient to simply enter the mapped drive id (e.g., "N:\"), as that drive mapping is also generally not available to services. Thus, one should specify the physical drives by letters, but must specify the network paths completely, e.g.,:
  M_BROWSE_PATH=C:\, E:\, \\network-drive\shared\

  Note that the above also applies even to script backup and restore drives.

- M_DATA_DIRECTORY to relocate the data such as the log files, and metadata incremental harvesting cache as needed for very large DI or BI tools

- M_JAVA_OPTIONS to increase the maximum memory used by java bridges during the metadata harvesting of very large DB, DI or BI tools. Note that this parameter defines the default maximum for all java bridges, however most memory intensive java bridges (e.g JDBC bridges) have the ability to define its own maximum memory in their last parameter called Miscellaneous.

8. User Interface Look & Feel Customization

8.1 Login and Headers

Customize the following files and directories using the embedded instructions (in comments):
   `%OMM_HOME%\conf\ressources\MM.properties`

%OMM_HOME%\conf\ressources\web\

8.2 Metadata Explorer for Business Users

Customize the following files using the embedded instructions (in comments):

%OMM_HOME%\conf\ressources\MetadataExplorer.xml