Yugandhar Open MDM (Master Data Management) Hub

* Providing Industry standard MDM solution for harmonization of Customer Master Data.
* In built and Extendable Customer Master Data Model for maintaining Person and Organization master data along with Account, Address, Phone Number, Identifiers, Grouping, Vehicle, Preference, Property and party relationships etc.
* Built using Open Source technologies – Spring boot, Hibernate and JBoss
* Built on Service Oriented Architecture (SOA) having 450+ prebuilt ready to use services
* RESTful and MQ enabled
* Rapid Development using Hibernate Reverse Engineering Tool and Yugandhar Code generation Templates (Freemarker).

Feature and Capabilities

1. Open Source and Completely Free, available with Apache 2.0 License.
2. Free and Open Data Model
3. 450+ prebuilt ready to use services, 50+ Composite and 400 Base table services
4. Embedded and extendable data validations and business rules
5. Multilingual Reference data management Capability
6. Comprehensive application and performance logging using logback.
7. User and role based Authorization
8. Inquiry level and Pagination for retrieve and search services
9. Wildcard capable search esrvic
10. Audit Log for maintaining database row level transaction history
11. Pluggable and UUID based Primary Key generator
12. jsr107 compliant Ehcache based caching framework
13. Integrated with Maven
14. Event Manakger
15. Match and Merge framework (Beta)

**About Yugandhar Open MDM Project**

Master Data Management came a long way in last decade or so. There are currently more than 20 MDM solutions catering to various specializations of MDM like Customer Data Integration (CDI), Product Information Management (PIM), vendor and supplier management etc. However most of these solutions come with licensing costs amounting to thousands of dollar. To offer a completely free solution which would be made available through Apache 2.0 license, A Project is started in 2017 under the name ‘Yugandhar Open MDM Project’ to build Open Source MDM solutions catering to CDI, PIM and Data Governance Capabilities. Yugandhar in Sanskrit means Ever Lasting and the strongest of its time. Our vision is to build the strongest, Open Source, Multi Domain, Cross Industry and completely free MDM Solution.

We are happy to announce that the first release of the Yugandhar MDM Hub catering to CDI solution is built with Open source technologies like Spring and Hibernate etc, inbuilt data Model, 400+ ready to use services and having incredible Out of the Box capabilities is currently being distributed. We aim to make the current CDI offering the strongest and Planning to bring Data Stewardship and PIM solutions in upcoming years.

Watch this space for the latest on Yugandhar Open MDM Project…

# Contacts

[**support@yugandharproject.com**](mailto:support@yugandharproject.com)

**support.yugandharproject @gmail.com**

# How To Contribute

If you have adequate knowledge of Open Source technologies and you are fit to contribute to this project then please feel free to send a mail to us at below email id mentioning the area in which you can contribute. We will respond to you at the earliest

[**support@yugandharproject.com**](mailto:support@yugandharproject.com)

**support.yugandharproject @gmail.com**

The Name of the contributors will be listed in either ‘Team’ or ‘Contributors’ or in both the section based on level of contribution

**System Requirement**

The supported configuration for hosting Yugandhar MDM Hub primarily depends on redhat jboss, Oracle and Springboot Framework configuration

Refer jboss System requirements here

<https://access.redhat.com/articles/2026253>

Refer Springboot System Requirements here

<https://docs.spring.io/spring-boot/docs/1.5.6.RELEASE/reference/htmlsingle/#getting-started-system-requirements>

**Production Environment**

We are summarizing the requirements for ease of access however for any further clarification refer the Redhat JBoss and SpringBoot framework Documentation

1. JBoss EAP 7.1 full runtime or later
2. Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production or Later
3. JBoss Integrated Artemis Messaging broker
4. Java jdk 1.8 or later

**Development Environment**

1. OS – Windows 7 enterprise addition, Service Pack 1 or later
2. 8GB RAM and 100 GB Storage
3. Eclipse Java EE IDE for Web Developers. Neon.3 Release (4.6.3) or later
4. JBoss EAP 7.1 full runtime or later
5. apache-maven-3.5.0 or later
6. Java jdk 1.8 (jdk1.8.0\_121) or later
7. Spring Boot 1.5.6.RELEASE
8. Hibernate Tools
9. Oracle Database 11g Express Edition Release 11.2.0.2.0
10. Oracle SQL Developer
11. SOAPUI or any other tool to test REST services

Note - Internet connectivity is needed to setup workspace. If internet is not available because of any reason then all the software and Maven jars needs to be manually downloaded which might be tedious task.

**Current Release and Downloads**

Version and Release Notes

Current Release – V1.0.0

Release Notes – provide download link

Git Repository – link

Java Doc – link

**Help and Resources**

Java Doc – link

Database Doc – link

Guide to setup – link

API reference guide – link (List of Out of the Box Transactions with sample xml)

Developer guide – link

**Developer guide index**

1. Yugandhar Open MDM Architecture
2. Understanding Java Projects and Workspace
   1. Project structure
   2. Java classes
      1. <test>Component.java
      2. <test>ComponentRule.java
      3. <test>Repository.java
      4. <test>Service.java
   3. Cache configuration
   4. Spring boot properties
3. Setting up the Workspace
4. Code Generator
   1. Generator for Data Tables
      1. <test>Component.java
      2. <test>ComponentRule.java
      3. <test>Repository.java
      4. <test>Service.java
      5. Artifacts.txt
   2. Generator for Reference tables
      1. <test>Component.java
      2. <test>ComponentRule.java
      3. <test>Repository.java
      4. <test>Service.java
      5. Artifacts.txt
5. Invoking the transactions
6. Understanding Data Model
7. Code Generation using Freemarker templates and Hibernate tools
8. Understanding different Components
   1. Data Tables and Reference Tables
   2. Extending existing Data Object
   3. Introducing new Data Object
   4. Modifying Business Rules
   5. Aspect Oriented Programming
   6. Authorization Framework (Access Control)
   7. Inquiry Level Framework
   8. Pagination Framework
   9. Reference data management
   10. Pluggable primary keys
   11. Application Logging
   12. Audit Logs
   13. Configuration Tables
   14. Match and Merge framework
   15. JMS Integration and Listeners

JBoss Reference link –

<https://www.redhat.com/en/technologies/jboss-middleware/application-platform>

Spring Reference link –

<https://projects.spring.io/spring-boot/>

Oracle ojdbc6.jar download

<http://www.oracle.com/technetwork/apps-tech/jdbc-112010-090769.html>

API reference guide - List of Out of the Box Transactions