The topic of this presentation is how to integrate IBM InfoSphere® Master Data Management Collaboration Server version 10 with other products to form a complete solution. IBM InfoSphere Master Data Management Collaboration Server is also known as MDMCS. This presentation discusses the product architecture, supporting software and other software which are optional but can make the application more robust and powerful.

Terminology

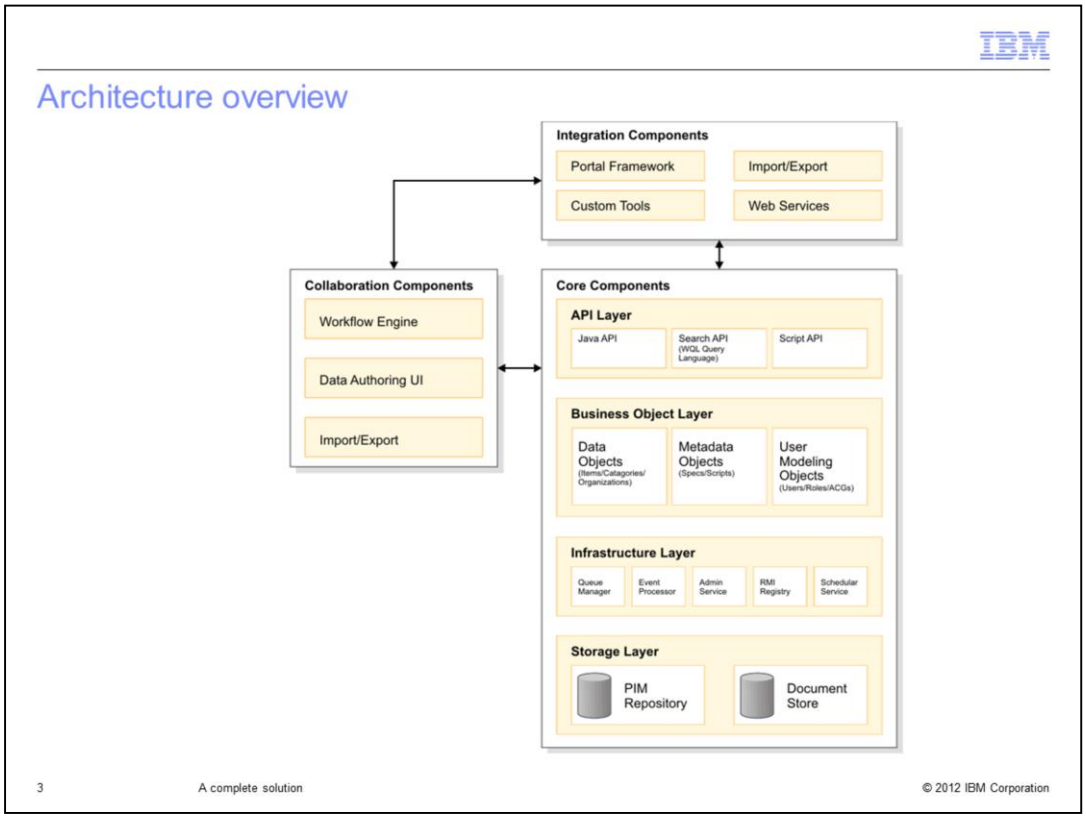
- Aim
  - Supporting software
  - Possible integrations
- Terminology
  - TOP
  - MDM
- What is not covered
  - Implementation guide

2 A complete solution © 2012 IBM Corporation

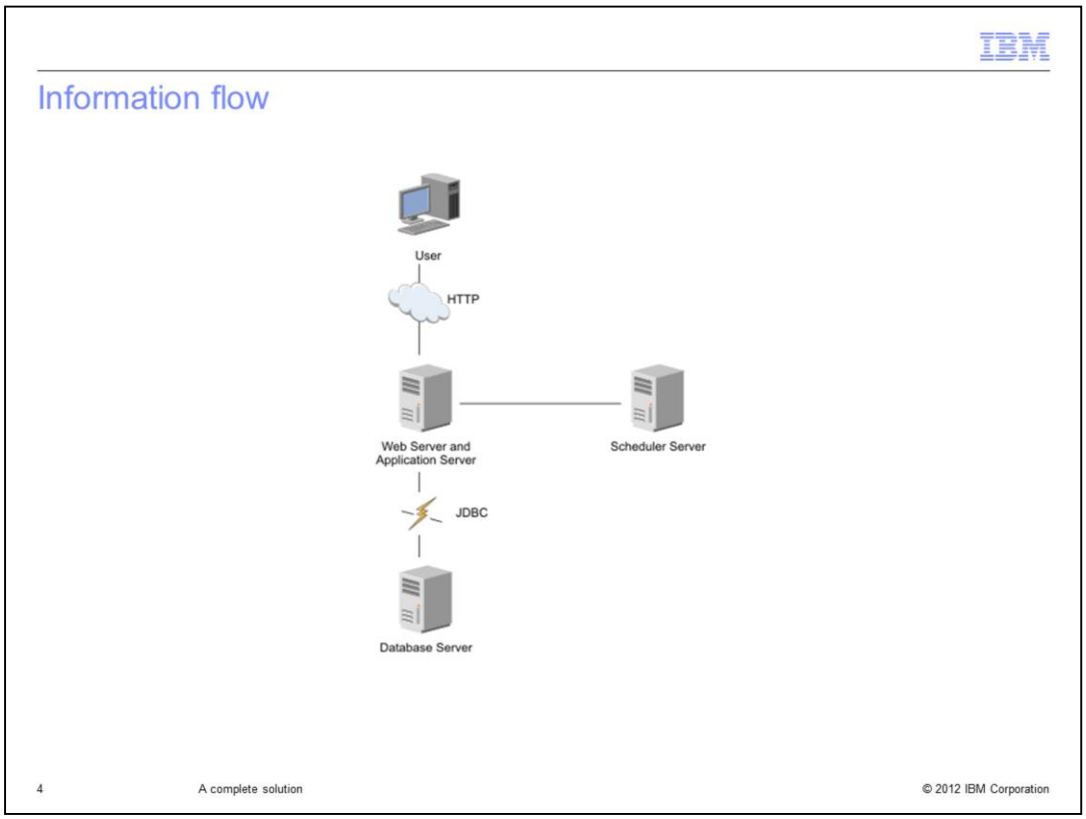
The objectives of this presentation are to discuss the supporting software and to discuss how you can integrate the software to develop a complete solution.

There is some terminology in this presentation you need to be aware of. The TOP variable is an environment variable that contains the location of the MDMCS installation directory. MDM refers to Master Data Management.

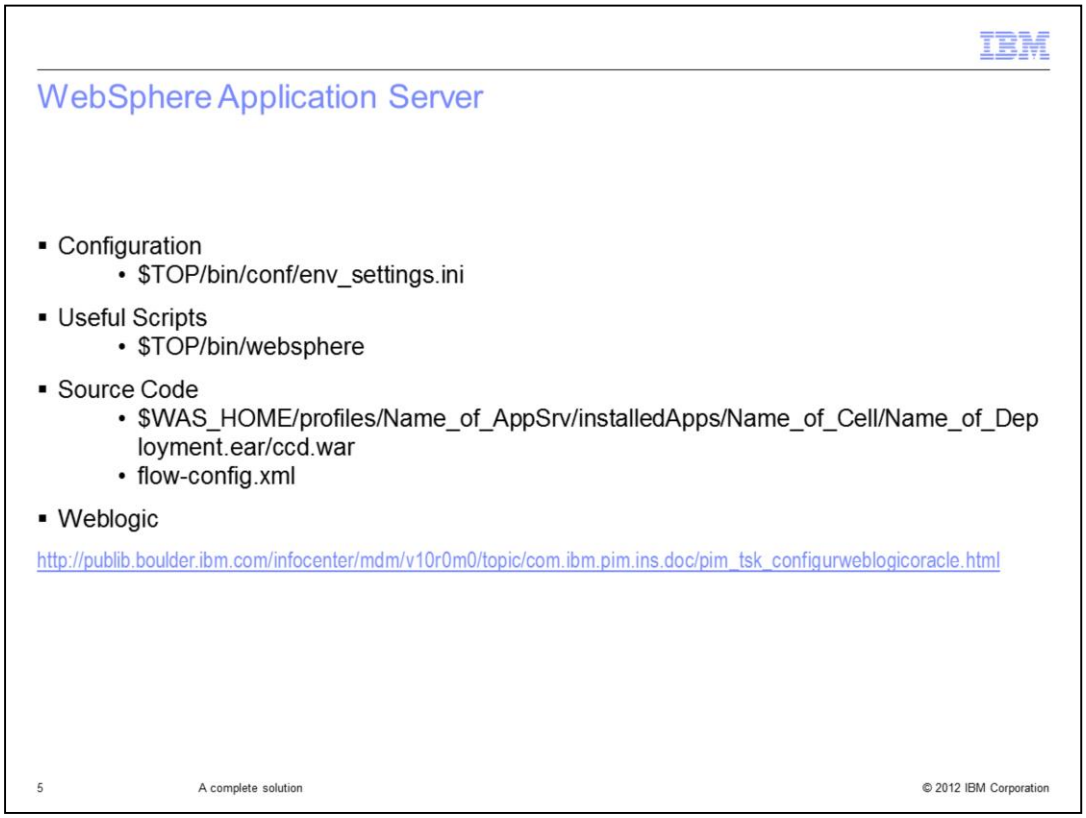
This presentation is not meant to be used as an implementation guide; see the Information Center or contact IBM Services for more information.



MDMCS has a component-based architecture that can consist of a two-tier or three-tier configuration. These components include: core components, integration components, and collaboration components. Visit the Information Center for details on these components.



The diagram displayed on this slide provides an overview of the flow of information between the client, web server, or application server, and the database server. There is also a scheduler service, which manages import and export jobs in the background. The salient features of this set up include an application server that handles HTTP requests from users. Services are started or stopped using RMI. The scheduler service uses the same RMI port as the one used to control services. Application and scheduler servers communicate with the database server using JDBC. Finally, the scheduler can be run on a dedicated machine or on an application server.

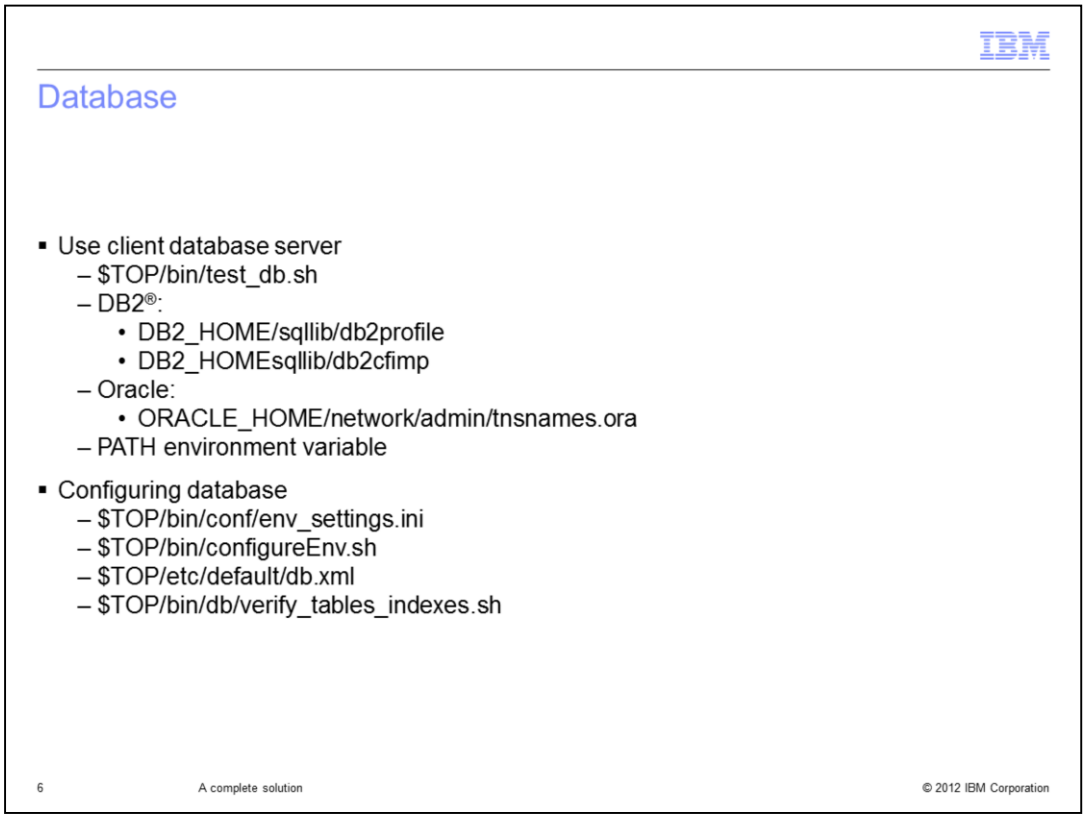


An application server is a necessity to run the application reliably. It is recommended you use the WebSphere Application Server because of its scalability, reliability, and security. However, IBM does support other application servers. WebSphere Application Server is configured in the $TOP/bin/conf/env_settings.ini file. The sections pertaining to the application server are appserver, appserver.appserver_type, appserver.appserver_name, and services.

After configuration, you need to deploy the application. For WebSphere Application Server, this can be done using the three scripts in the WebSphere Application Server directory; create_vhost.sh, create_appsvr.sh, and install_war.sh. These scripts are designed for the stand-alone version of WebSphere Application Server and as of version 10.0 FP2, they do not work with a clustered network deployment of WebSphere Application Server.

The application source code is compiled in the form of .class files and is governed by the application server. To locate the code, follow the directory structure displayed on this slide. The flow-config.xml file is the core configuration file for the framework and should not be modified except in rare circumstances like deploying custom tools. Running the $TOP/bin/websphere/install_war will redeploy the application server and erase all customizations.

The product also supports Weblogic. Configuring Oracle WebLogic Server is a two-step process and you must create a server domain and a domain component. Refer to the Information Center link displayed on this slide for details.



Like application server, a database is also essential for running the application. Both application server and database server are resource intensive and they should be installed on different machines. To do so, set up the client server with the database server such that the instance user and the database schema can connect to each other. To test the connectivity, use the $TOP/bin/test_db.sh script.

To configure DB2, you need to import the database server information in the client using the db2cfimp command. You also need to add the location of db2profile in .bash_profile or .bashrc file of the MDMCS application user. This will set all the environment variables needed like DBINSTANCE. Similarly in Oracle, you need to include the database server details in the tnsnames.ora file in the ORACLE_HOME/network/admin directory.

For both DB2 and Oracle, the classpath variable should include their respective installation location.

The database itself is configured in the db and db.db_name sections of the env_settings.ini file. The changes take effect after you run the $TOP/bin/configureEnv.sh which builds or overwrites the db.xml. It is from this db.xml that the run time parameters are set to establish a connection with the database.

There is also a verify_tables_indexes.sh script which checks the database schema for consistency.

LDAP is a set of protocols for accessing information directories and is commonly used for managing users. Integration of LDAP with MDMCS enables your system to support over 1000 casual users where each user requires authorization for various internal and external roles. Some of its advantages include real-time LDAP user entitlement, synchronization and ease of maintenance with a central LDAP server repository, quick setup, and a single source of truth for user records.

You can enable LDAP by setting the "wpcOnlyAuthentication" variable in Login.wpcs script to “False”. After this, configure your LDAP server, create user and group records, and then match the MDMCS records with the corresponding LDAP records. Follow the Information Center link displayed on this slide for a detailed explanation of this process.

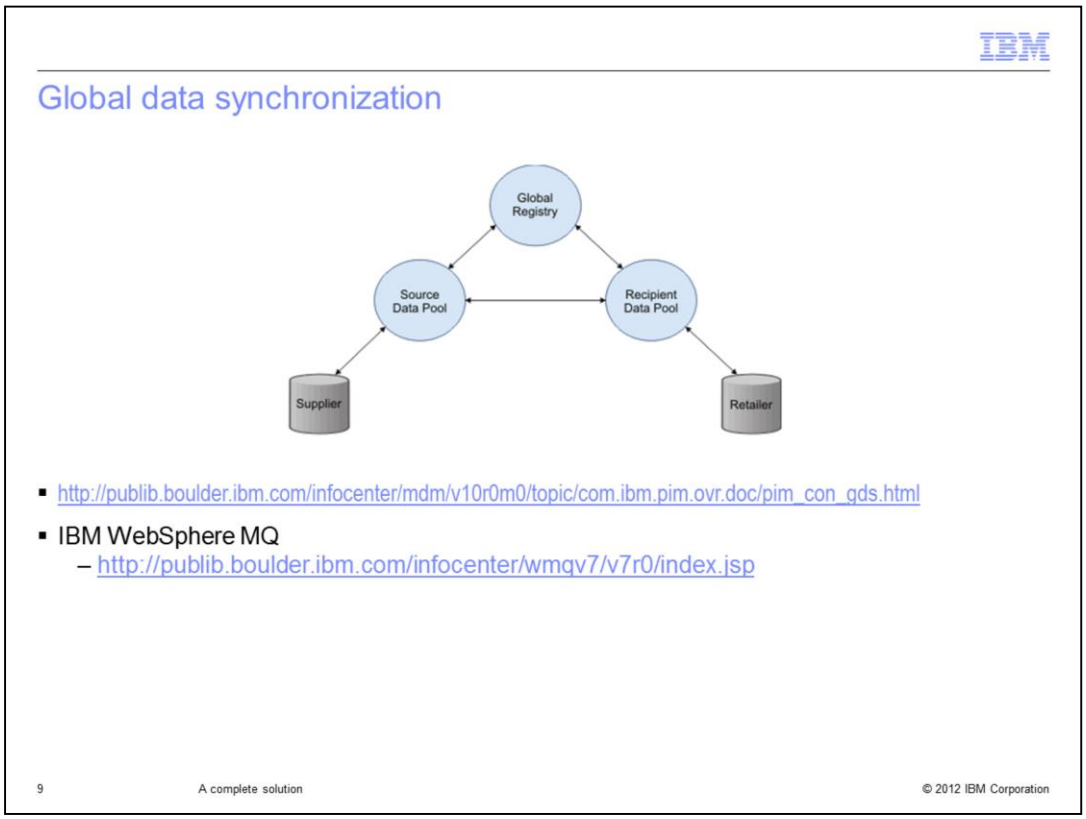
A web service is a software system designed to support interoperable machine-to-machine interaction over a network. Web services can convert your applications into Web-applications which can enable one application to read and modify data of another without compromising security. Web services are governed by global standards of implementation SOAP. One such standard is Apache Axis 1.4 and currently it is the only standard supported by MDMCS.

To enable Web services, you must define a value for soap_company, soap_user, and product_center_url, in the common.properties file. Incoming requests will use this company and user to access the database and run scripts. Running the $TOP/bin/configureEnv.sh will over-write any customizations to the common.properties file.

Use the Web services console to manage Web services. At this console, the various Web services are shown as rows in a table which makes it very convenient to mange or to invoke them.

The product is bundled with several Web services which are located in the $TOP/samples/webservices folder. Additionally, there is a Web project that contains implementation of these Web services and may be used for customizing them. This will give you a bottom up implementation approach.

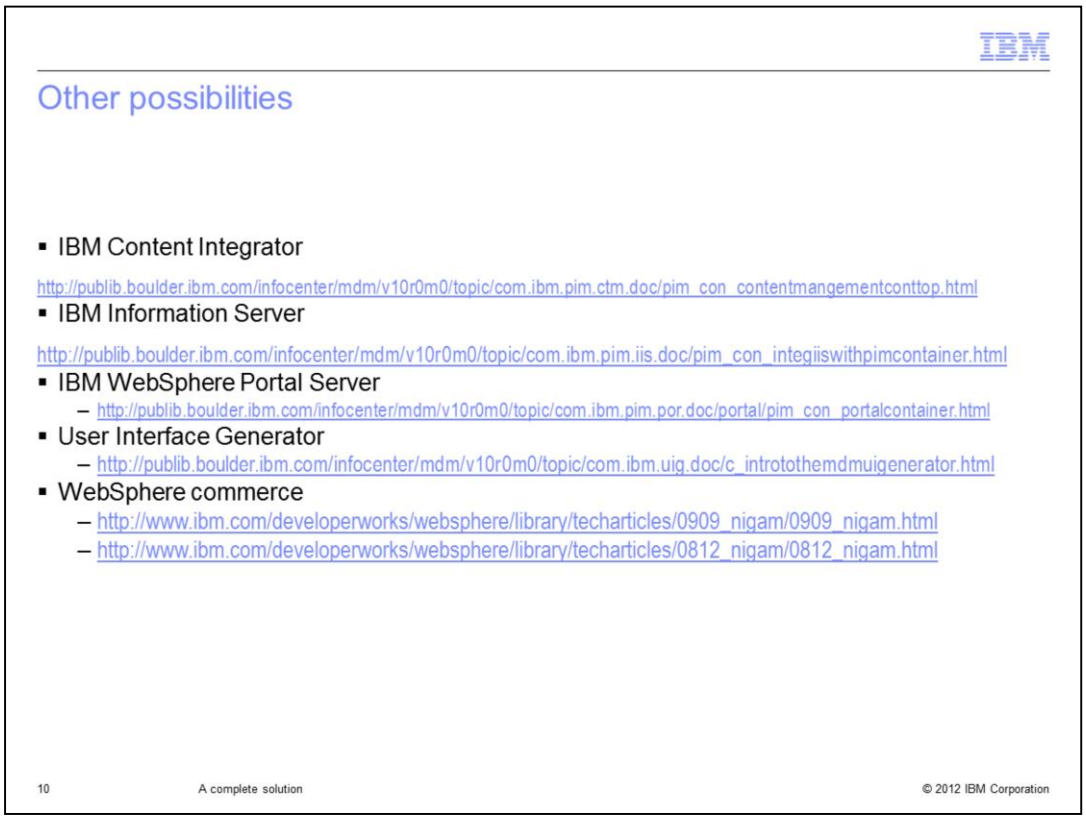
During implementation and maintenance of MDMCS, you will have the need for an integrated development environment for implementing custom code, for example, scripts, custom tools, Web services and so on. Although Eclipse is sufficient for most of these tasks, it is suggested you use the Rational Software Architect, or RSA. Visit the links displayed on this slide for details. The first link is to the homepage of the RSA Information Center and the second link has details on how to set up and configure RSA with MDMCS.



Global data synchronization, referred to as GDS, is an IBM offering which is bundled with MDMCS. GDS is an ongoing business process that enables continuous exchange of data between trading partners to ensure synchronized information. It is based on a publish or subscribe model. The supplier is required to publish its product information to a data pool, and the data pool then matches the published data to the known subscribers of the data.

The product information can be about price, party, and other relationship-specific attributes. This process ensures that all stakeholders for a product are notified about the latest information about the product. Follow the Information Center link displayed on this slide for details on this offering.

To deploy GDS, you must install IBM WebSphere MQ, which is a system for messaging across multiple platforms and is a key component in IBM's service oriented architecture. Its benefits include independence from packet-based transmissions, like TCP/IP and reducing redundancy; messages will only be delivered once irrespective of errors and network problems.



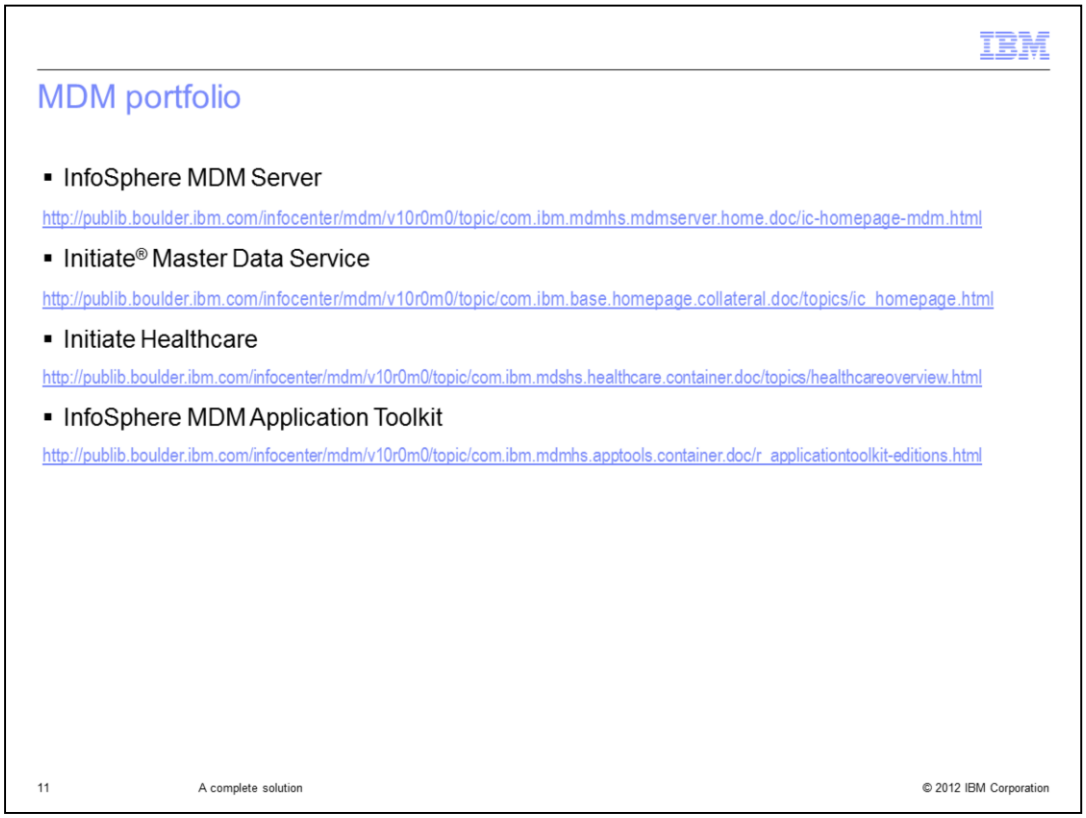
This slide displays some other optional software which can be used with MDMCS. The IBM Content Integrator works as the interface between MDMCS and any content management system. Content management systems often store unstructured data, such as product images, specification data sheets, warranty documents, and demonstrations, which provide relevant and necessary contexts for products.

IBM InfoSphere Information Server is a market-leading data integration platform that helps you understand, cleanse, transform and deliver trusted information to your critical business initiatives. MDMCS provides Java APIs that can be used to export PIM meta data, such as, specs and attribute collections for use in IBM Information Server tools like DataStage® and Business Glossary.

IBM WebSphere Portal is a powerful solution for delivering web content and applications in an integrated, differentiated and personalized web experience. WebSphere Portal supports workflows, content management, social media, mobile web delivery, simplified usability and administration, open standards, security and scalability. The InfoSphere MDMCS and WebSphere Portal Server solution, is ideally suited as a self-service application, where you can configure the solution so that vendors can enter and manage their own product information.

The User Interface Generator is an Eclipse tool that takes a user model and generates a role-based user interface for a Java web application. It enables you to rapidly generate user interfaces using industry standards such as UML and Java EE.

WebSphere Commerce offers complete "sell side" business-to-business solutions for today's e-businesses. Visit the two DeveloperWorks article links displayed at the bottom of this slide to learn more about integrating WebSphere Commerce with MDMCS.



This slide contains links to the Information Center of other IBM InfoSphere Master Data Management offerings.

Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, DataStage, DB2, InfoSphere, Initiate, Initiate Master Data Service, Rational, and WebSphere are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2012. All rights reserved.

© 2012 IBM Corporation