Dimensions RM—Scaling for the Enterprise

Dimensions RM Product Team
Micro Focus
<table>
<thead>
<tr>
<th>Table of Contents</th>
<th>page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>1</td>
</tr>
<tr>
<td>Architecture</td>
<td>2</td>
</tr>
<tr>
<td>Deployment</td>
<td>8</td>
</tr>
<tr>
<td>Scalability</td>
<td>10</td>
</tr>
<tr>
<td>Summary</td>
<td>16</td>
</tr>
</tbody>
</table>
Overview

Dimensions RM is the leading requirements management solution with the most advanced and comprehensive set of tools in support of requirements capture, verification, visualization, organization and prioritization. Highly integrated into the development lifecycle, Dimensions RM provides a streamlined requirements process from business needs to development and testing. With its comprehensive traceability capabilities and requirements change control, Dimensions RM dramatically improves the visibility within the development lifecycle and the quality of the resulting products.

Dimensions RM delivers:

- End-to-end traceability, from the inception of a project or issue through delivery of the functionality
- Visibility and control over project status, progress, and prioritization
- Efficient requirements collaboration between stakeholders
- Unified change process & Streamlined Impact Analysis
- Multi-modal support for waterfall, agile, and hybrid approaches
- Requirements-based Testing

This document describes the high-level architecture of Dimensions RM highlighting the major components of the system and how to deploy them in an enterprise environment to ensure your Dimensions solution scales and performs well.
Architecture

The following diagram gives a general overview of the tiers in the Dimensions RM architecture including which components reside in each tier and how they connect:

As you can see, Dimensions RM has an n-tier cross-platform architecture consisting of the following layers:

- Presentation tier
- Business logic tier
- Database/Metadata tier

The components in each of these tiers can be installed all on a single system or onto separate physical machines. Because the components can be scaled horizontally (spread across a number of machines) to expand capacity, Dimensions RM can grow to meet the needs of your enterprise.

**Presentation Tier**

The presentation tier consists of the clients that end-users will make use of to access the functionality of Dimensions RM. There are a number of different clients to allow support for different access methods.
Components within the business logic tier are server components responsible for carrying out Dimensions RM operations on behalf of the clients.

Below is a detailed diagram of this tier.

- **Web-based Client**
  The Web Client provides a web-based interface to the end-user features of Dimensions RM. This client uses the standard HTTP (or HTTPS) protocol to communicate to a Web Application Container (aka servlet container) in the business logic tier. The use of browser-based HTTP communication enables deployment of these clients across a wide variety of platforms and network topologies.

- **Import Client (RM Import)**
  The Import Client (RM Import) provides a Windows fat-client front-end (.NET based) for performing import activities from external tools MS Word and MS Excel.

- **Administration Desktop Client (RM Manage)**
  The Administration Client (RM Manage) provides a Windows fat-client front-end for performing tool administration and configuration activities.

**Business Logic Tier**

Components within the business logic tier are server components responsible for carrying out Dimensions RM operations on behalf of the clients (e.g. authenticating users, checking privileges/authorization, enforcing workflows, maintaining metadata about products, projects, baselines, requirements, etc.).

These server components are supported on Windows platforms. See our web site or the Readme file for a list of the latest supported platforms.
The Dimensions RM Server component is where the main business logic of Dimensions is performed.

---

**Web Application Container (Dimensions RM Web Application)**

The Dimensions RM Web Application is a J2EE application running in an servlet container (such as Apache Tomcat); this component deals with requests from the Web Clients and Import clients.

The Web Client is built using Java, HTML, and JavaScript, JSP (Java Server Pages) technologies.

Most of the product’s business logic is implemented by the Dimensions RM Server. The Dimensions RM Web Application communicates over pipe to the Dimensions RM Server to perform operations, query data, store/retrieve file content etc.

When the product is installed it is deployed and configured to run under Apache Tomcat, but it can also run under other supported Web Application Containers (see product release notes for details) and the roles of the web server and servlet container can be separated, e.g. Microsoft IIS (Internet Information Services) could be used as the web server and Tomcat for the servlet container. By default the web applications are hosted by Tomcat on TCP/IP port number 8080 but again this is configurable.

**Dimensions RM Server**

The Dimensions RM Server component is where the main business logic of Dimensions is performed. This component is implemented using a proprietary C++ application management framework that provides high scalability and performance and has been proven in many enterprise environments over the years.

There is a Pool Manager component that manages a pool of Dimensions RM application servers and re-starts the server processes in the event of a failure giving the system maximum availability. When a user is actively using a Dimensions RM client they are allocated a process from the pool, when they become idle the process returns for the pool to become available for other users. The initial, minimum and maximum size of the pool can be configured. This pooling architecture allows high scalability as many users can be connected to Dimensions RM but only the active ones are taking up precious system resources.

The Dimensions RM Application Server process is where the main business logic is performed; these processes connect to the database to perform queries/updates and communicate with the client. These processes that are “pooled”. 

Below is a detailed diagram of this tier.
Dimensions RM is a highly configurable tool with a rich set of relationships and properties of the assets being managed, so to enable high performance and scalability Dimensions RM uses relational database technology to store and query this metadata.

The Pool Manager must be run as a Windows service using the Windows System account or an account with similar access rights.

- **Serena License Manager**

  The license manager is basically an extension of the Macrovision FlexLM licensing product; it supports both concurrent and named user licensing models. In addition to the license server, Serena also provides a Java Swing based GUI to perform license administration activities.

  When a connection is made from a client, Dimensions RM Server calls the license manager to obtain a license for the client. When the user exists his client or a session timeout (configurable in length) occurs the license is returned. Communication between the license manager and Dimensions server is via TCP/IP using Macrovision's own APIs and protocols.

**Database/Metadata Tier**

Dimensions RM is a highly configurable tool with a rich set of relationships and properties of the assets being managed, so to enable high performance and scalability Dimensions RM uses relational database technology to store and query this metadata.

Below is a diagram of this tier.

As you can see, to support enterprises Dimensions RM supports Oracle databases. For a list of the latest supported database versions, check our web site or the Readme file for the version you are evaluating. For connection with Oracle databases Dimensions RM uses the Oracle Call Interface API (OCI).

A single Dimensions server can connect to multiple database instances, within an instance there can be a number of Dimensions RM databases (for example perhaps one database instance is used for your production work and another one for testing).

Tools are provided to carry out routine database maintenance and monitoring and documentation is provided covering backup and recovery of the database.
Platform Support
Dimensions RM has broad platform support allowing the use of Windows, Linux, many flavors of UNIX and z/OS for the IBM Mainframe. See the Dimensions RM Readme document for further details on supported platforms.

Application Programming Interface (API)
Dimensions RM offers a publicly available supported API (Application Programming Interface) allowing customers to create their own applications that make use of Dimensions RM functionality or to extend the functionality of Dimensions RM.

- Web Services
  A comprehensive set of SOAP and REST Web Services are provided with Dimensions RM to perform most tool operations. The Document Literal Wrapped style of WSDL declaration is used and the WSDL is WS-I Basic 1.0 compliant.

  In the diagram below a web service client is using some kind of SOAP Client API to send the SOAP request over HTTP to the Web Application Container. This could be any SOAP 1.1 compliant client. In the business logic tier the Dimensions Web Services are implemented using Apache AXIS so the AXIS servlet receives the request and calls into the Dimensions RM web services code. The web services call the Dimensions RM server using TCP/IP in the same way as the other Dimensions RM web applications.

- URL addressable Views
  In addition to using programmatic interfaces to access Dimensions RM data, you can also use URL addressable views to access requirements, reports, containers, etc.
Security
When communicating between tiers (across the network) Dimensions RM employs various strong encryption technologies to ensure sensitive data is transmitted securely.

- **HTTPS**
The Web Client and Import Client can be used over the industry standard HTTPS protocol to encrypt all communication between the browser and the Web Application Container using SSL/TLS.

- **LDAPS**
When using LDAP to authenticate user credentials Dimensions RM can support both LDAPS (LDAP over a SSL tunnel) and LDAP with StartTLS extensions. Both of these security mechanisms encrypt the LDAP communication using strong SSL/TLS encryption. Not all LDAP servers support both secure protocols so choose the relevant one for your particular server.

Authentication
When a user invokes a Dimensions RM client they must login before using the tool, a user name and password must be provided. The Dimensions RM Server needs to validate the user name and password are correct and it can be configured to do this using a number of different authentication systems.

Native Dimensions RM Authentication
This is the default configuration for authentication and uses the Dimensions RM database to store and validate the users credentials.

LDAP
The Dimensions RM Server can be configured to authenticate users using a LDAP (Lightweight Directory Access Protocol) server. Some examples of LDAP servers are Microsoft Active Directory Server, the Java Directory Server from Sun Microsystems and the OpenLDAP directory server. Enterprise customers often use these directory servers to centralize their user account information.

SSO
The Dimensions RM Server can be configured to authenticate users using a SSO (Single-Sign-On) server. SSO server examples are SBM, Dimensions CM or Microsoft Azure AD.

CAC
The Dimensions can be configured to authenticate with CAC (Common Access Card) providers to support card based security. This is only available with SSO usage.
Authorization
Each Dimensions RM operation is controlled by a privilege; these privileges have a set of rules which (if met) grant the user permission to carry out the operation. These privilege rules are very flexible allowing control over who can do what based on the state of the object being operated upon, the groups the user is a member of, the roles that the user holds etc.

In addition to the privilege other process rules are also evaluated to see if the operation can be performed.

User Groups are an important aspect of the Dimensions RM authorization mechanism, user groups can be used in conjunction with privileges to control access to assets.

See the Dimensions RM Administrator's Guide for further details on how Dimensions RM privileges, users, categories and group function.

Deployment
There are a wide variety of configurations that are supported by Dimensions RM. The next set of diagrams shows Dimensions RM deployed in a few common configurations.

- Standalone Server
- Multiple Servers

They show how the architecture that as been described can be deployed onto physical systems. The components are designed to be installed and configured easily. Once you have chosen to deploy your initial system, it is possible to change and expand as your needs grow.

Standalone Server
In this configuration the components from the lower 2 tiers of the Dimensions RM architecture (business logic and database) are all hosted on a single physical server machine. This may work well if you have a small user community or a powerful server machine.
Multiple Servers

In this configuration, the database has been moved off onto a separate machine from the Dimensions RM server. This spreads the CPU and RAM needs of the server components across the two machines allowing for better performance and scalability.
Scalability

As you have seen, Serena Dimensions RM is architected to scale to a variety of Enterprise needs, but providing a scalable environment for enterprise applications is complex. Dimensions RM has many possible configurations. Many factors must be closely considered including network hardware and software, WAN integration, server hardware and software, network load, server load, and more.

Prior to deploying Dimensions RM, these factors should be considered by your IT professionals and if needed with advice from Serena Software Professional Services. Your data and usage models will need to be considered as well when determining how to scale Dimensions RM to meet your needs.

Vertical Scaling
Scaling up involves applying higher powered hardware on the Dimensions RM server. This can involve faster CPUs, multiple CPUs, more memory, faster Network cards, or more likely, some combination of all of these. Although the information shared in this section applies to scaling up a system with the entire application installed on it, it is also applicable when choosing to scale horizontally and should be considered there as well.

Processors
Depending on the operations being performed, the Dimensions RM server and the database can be CPU intensive. When using multi-processor or hyper-threading configurations, the server is able to use more system resources than under single processor configurations. In memory intensive situations, the server may even consume all available memory. Having two gigabytes of memory available per processor (virtual or physical) is a good general rule of thumb.

Memory
Memory is important to scaling Dimensions RM. There are various active Dimensions RM processes that require memory. Making sure there is available memory when needed will allow you to scale further.
Dimensions RM Server memory
The RM Server can be scaled by adding memory. The processes of the RM server were described earlier. The “application server” process's (rmAppServer.exe) can be controlled by adding command line parameters to RMServerPool service (see User documentation for more details). Here you can control the number of active processes and how they startup. One of these processes is needed by each active/concurrent connection you plan to support. Starting more initially and leaving them running to service user connections will consume more memory, but will reduce the amount of CPU and other processing activity involved in starting the process when they are not readily available. For each active/concurrent user you plan to support, you can estimate using approximately 48MB for each; minimum memory size is 200MB for a process. By adding more memory on the RM server, you will find the ability to support additional “application server” processes is increased and will allow your system to handle more user connections efficiently.

File Storage Server memory
The “application server” processes require some file storage memory for logging information. Also File Storage system is used as cache for images which can be attached to RM objects and for import/export operations. Approximately 50MB will be enough for each user.

Web Application Container memory
Our default installation installs a java based Web Application Container (Apache Tomcat). Adding more memory to the Application server can improve your ability to scale vertically and support additional web clients. The amount of memory that is used is controllable via configuration settings, which can be located in our documentation. Since we do also support other Web Application Containers as well, you should consult their documentation for details on expanding memory if you are using one of those containers.

Database server memory
Making sure your database has ample memory is important for scaling your system. Simply adding more memory may not necessarily allow the database to properly utilize it in all cases. We recommend that you consult Oracle database documentation for how to optimize the memory and configuration. As a general rule, for each additional active/concurrent process approximately 20MB is consumed. You should consider the type of activities that your user population will be performing when sizing this area. For example if you expect many people to be constantly running reports, you will want to make sure there is enough memory to support those processes and the database is tuned properly to take advantage of it.
Storage Areas
The disk subsystem in a single system configuration can also be used to get additional scalability. Generally the faster the disks are, the faster your operations will be performed. If you have the ability to use multiple drives, doing things such as separating the database data and the database logs on to separate physical will allow the system to operate more efficiently. If possible place the operating system on a drive of its own. The more you can reduce the contention for data access off the physical disk drive, the more scalable the configuration will be. There are also techniques such as disk striping that can assist in increasing the throughput in certain disk configurations.

Networking
The systems' networking is an important aspect of scaling. If the network card is saturated with network traffic in your configuration, an additional network card can be used to help scale the application.

Horizontal Scaling
When it comes to achieving the highest levels of concurrency, scaling out horizontally becomes necessary. Simply moving the components to separate machines is a great start when attempting to scale horizontally. Since the Database and RM Server will most likely be contending for CPU, RAM and disk I/O, it is recommended that you separate these pieces first.

When you begin to apply a horizontal scaling strategy, you will have further networking considerations. Consider keeping the RM Server and DB Server on the same network subnet if possible. The closer in proximity the servers the less impact you will see due to network traffic/delay.

To extend the capacity of the system you essentially have the option of doing some form of load separation and load balancing.

Dynamic Load Balancing
The Web Application will be able to take advantage of a web farm that contains multiple Web Application Containers (e.g. Tomcat). The farm of Web Application Containers can be implemented using either hardware or software. Hardware web farms require specialized hardware and skills to implement and tend to be more costly. A software web farm can be implemented completely using Microsoft Windows Server. The network load balancing capabilities of Windows Server automatically distribute the load across servers in the farm. In either type of Web Farm, the end user continues to access the system via a single URL and is completely unaware of the Web Farm implementation.
Specifically the Web Application Container can be load balanced using usual techniques: DNS based (with the known drawback of the DNS Caching mechanism) or Reverse Proxy based (whether using software techniques as Squid Internet Object Cache, Netscape’s or Microsoft’s Proxy Server, and Sun’s Netra Proxy Cache Server or using hardware techniques as Cisco Systems’ LocalDirector and Coyote Point Systems’ Equalizer).

The Dimensions RM server can also be load balanced using a hardware load balancer that supports the use of “server affinity” algorithms. A Dimensions RM client needs to connect to the same physical hardware for the Dimensions RM Server after an “idle” time, this means that the client needs to communicate to the same IP address for the server for the lifetime of its session.

If you wanted to expand the capacity of your RM servers to support additional Web and Desktop clients, your system deployment might look similar to this next diagram.
Static Load Balancing
The nice part about dynamic load balancing is that all the Web Farms are seen to the end users as a single system even though they are made up of multiple systems. If you are not able to implement dynamic scaling, you can still scale the deployment manually.

Each Dimensions RM Servers would be set-up and linked to the central database. Each Dimensions RM server will hold part of the load of the users connecting to Dimensions RM. There will be no dynamic load balancing happening and you will have to define a set of fixed users to connect on each of the server that you will set up. You might do this by instructing different departments of your organization to access one of the specific systems that had been designated for them.

Configuration Guidelines
Dimensions RM performance/scalability can be most impacted by the following factors:

- Network Configuration/Topology
- Server CPU speed
- Serena Dimensions RM Configuration
- Usage Model

Network Configuration/Topology—Dimensions RM is distributed across the clients, servers, and database server. Consequently, network hardware and software configuration and capabilities have a significant impact on the performance characteristics of Dimensions deployment. The closer in proximity on the network these pieces are the better performance you will see. As a general rule of thumb, you should attempt to have your servers and database tiers on the same network segment if possible.

Dimensions RM CPU speed—The server and database transactions can be CPU intensive. Faster CPUs translate directly to improved server performance. Do not under invest in the quality and speed of the server CPUs. The more processes that are running on a single machine, the more the CPU demand will be once additional load is applied.

Software Configuration—Dimensions RM's configurability, while being one of the products greatest advantages, can lead to performance issues if not implemented in accordance with best practices. Serena's technical support and professional services teams are well versed in these best practices and should be engaged to review implementations and help diagnose performance issues.
Usage Model—You can imagine that there are many possible usage models and of course each one will utilize systems resources in a unique way. Understanding the usage model for your user base will allow you to determine how to best scale the servers to meet your company’s needs.

A General Approach for Considering Your Needs

When optimizing performance, focus first on getting servers with the fastest possible CPUs. Second, look closely at memory utilization and insure there is enough. Then, consider multiple processor configurations. These principles should be applied to servers when both scaling vertically and scaling horizontally. In the end, any single Dimensions RM operation happens fastest when CPU speed, network bandwidth, and available memory are high. Using the fastest possible CPU at the client, server, and database tiers result in the best end user experience.

![Dimensions RM Scalability Zones Chart](chart.png)

The chart at left illustrates scalability zones for Dimensions RM.

1. Single system with all components: 4-core CPU, 8-12 GB RAM
2. Single system with all components: 12-core CPU, 16-24 GB RAM
3. Scale up additional CPU/RAM or separating RM server and database server:
   - RM server: 12-core CPU, 16-24 GB RAM
   - Database: 16-core CPU, 24-32 GB RAM
4. RM server farm and database (cluster):
   - RM servers: 6 cores + 0.1 x # of users, 6 GB + 48MB x # of users RAM
   - Database servers: 6 cores + 0.15 x # of users, 6 GB + 64MB x # of users RAM
   - Total numbers for entire server farm
In this chart, there are four zones. Zone 1 represents low concurrency and simple configuration and usage patterns. Zone 4 is highest end of the spectrum. It represents high concurrent user load and complex data and usage patterns.

This table can be used to determine appropriate server configurations based on usage patterns and expected number of concurrent active users. The configuration recommendations are the results of analysis of data collected as part of the rigorous Dimensions RM acceptance and certification process.

A concurrent user is defined by a user in the system who is actively performing work (e.g., running reports, opening requirements) within a certain cycle. More users can be logged into the system but are not considered concurrent until they are performing some activity. With this definition you can determine your usage pattern and the number of concurrent processes that you anticipate to consider a hardware configuration that might fit your needs. If you want for example to support a population of 250 users and you believe you will have 50 users performing concurrently moderate operations, then Zone 2 would be the starting point for the hardware you might consider. If you had a 1000+ user population to support, Zone 3 at 100 concurrent users might be appropriate for you depending on the actual usage.

Summary

Serena Dimensions RM has been architected, implemented and tested to meet the complexity and scalability challenges of the enterprise. Its meta-data driven design provides extreme configurability and flexibility. Its open architecture provides several solutions to enterprise data connectivity. By leveraging state-of-the-art technology, Serena Dimensions RM provides a platform that will grow to meet the ever expanding needs of the enterprise.