

Administrator's Perspective of SAP NetWeaver MDM – Part 2



Applies to:

SAP MDM 5.5 SP06, SAP MDM 7.1

For more information, visit the [Master Data Management homepage](#).

Summary

This is the second part of the series of articles describing the various aspects of Administration of the SAP NetWeaver MDM application from a System Administrator's perspective.

This particular article focuses on the "Sizing" dimension which is crucial to ensuring the optimal utilization of an application.

Author: Neelesh Thackeray

Company: Mahindra Satyam

Created on: 06 August 2009

Author Bio



Neelesh has 20+ years IT experience including 12+ years in SAP. He has been working on SAP NetWeaver MDM at Mahindra Satyam since March 2005 in the capacity of COE Leader. He has also been involved in Strategic due-diligence study for MDM at various clients as well responsible for competency development activities in SAP MDM.

Table of Contents

Introduction	3
MDM Architecture	4
Key Terminologies	5
What is Sizing?	6
Why is sizing important?	6
Summary.....	8
Related Content.....	8
Disclaimer and Liability Notice.....	9

Introduction

In the [previous article](#) (part 1), we have seen the importance of **SysAdmin** including the role / responsibilities that form part of the job profile.

In the context of the SAP NetWeaver MDM application, the administrator has added responsibility to ensure that the application is up and running 24 x 7 x 365 days. SAP MDM plays a key role as a central data hub providing cleansed master data to the business processes running on disparate systems.

This particular article focuses on the next important aspect – “**Sizing**”. The elements that constitute “Sizing” are as shown in the diagram below:

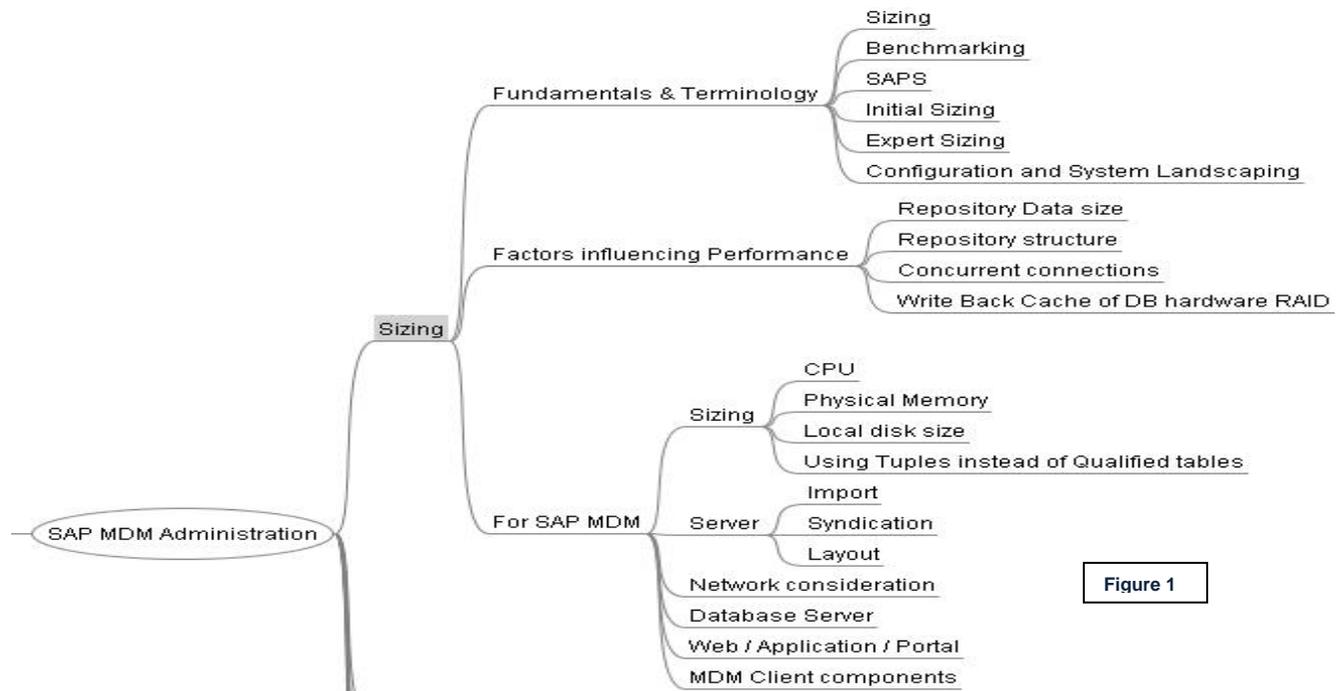
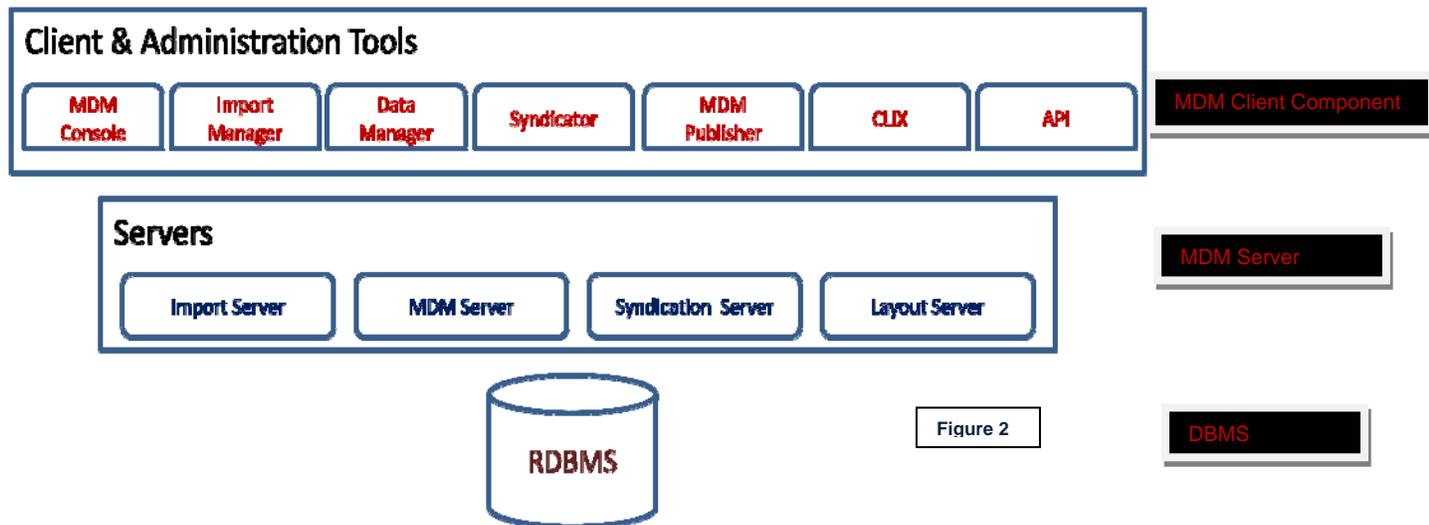


Figure 1

Note: This article is NOT a replacement for any SAP documents provided on this topic, although some information has been brought in from these documents. The emphasis is primarily to understand the concept of “Sizing” with answers to “W” questions, namely: What, Why, etc in general and then relate it to the hardware sizing requirements for installation of SAP MDM application.

MDM Architecture

Before we deep dive into the area of “Sizing”, let us revisit the SAP MDM architecture which is as shown below:



As can be seen above, the SAP MDM application is a suite of components broadly categorized into following groups namely;

1. MDM Servers
 - a. MDM Server
 - b. MDM Import Server
 - c. MDM Syndication Server
 - d. MDM Layout Server
 - e. RDBMS
2. MDM Clients
 - a. MDM Data Manager
 - b. MDM Import Manager
 - c. MDM Syndicator
 - d. MDM Publisher
 - e. API
 - i. ABAP API
 - ii. Java API
 - iii. .NET API
3. MDM Administration Components
 - a. MDM Console
 - b. MDM CLIX

Typically, SAP MDM application components mentioned above will be installed on various machines including desktops / Laptops and “Server-class” machines depending on the types of users and their roles. For example, the MDM console and MDM CLIX is typical used by the SAP MDM **Sysadmin** and Data Modelers (for MDM repository management), the MDM clients will be used by other business users with the MDM Servers installed on “Server class” machines.

There could be additional installations for staging and/or sandbox applications depending on the customers’ IT policies and business practices.

Key Terminologies

The key terminologies that need to be understood which have a big impact on the SAP MDM application installation strategy and operational use are as follows:

i. Load Balancing

As per Wikipedia:

In computer networking, load balancing is a technique to spread work between two or more computers, network links, CPUs, hard drives, or other resources, in order to get optimal resource utilization, maximize throughput, and minimize response time. Using multiple components with load balancing, instead of a single component, may increase reliability through redundancy. The balancing service is usually provided by a dedicated program or hardware device (such as a multilayer switch).

ii. Failover

As per Wikipedia:

Failover is the capability to switch over automatically to a redundant or standby computer server, system, or network upon the failure or abnormal termination of the previously active server, system, or network. Failover happens without human intervention and generally without warning, unlike switchover.

Systems designers usually provide failover capability in servers, systems or networks requiring continuous availability and a high degree of reliability.

iii. High Availability

As per Wikipedia:

High availability is a system design protocol and associated implementation that ensures a certain absolute degree of operational continuity during a given measurement period.

Availability refers to the ability of the user community to access the system, whether to submit new work, update or alter existing work, or collect the results of previous work. If a user cannot access the system, it is said to be unavailable. Generally, the term downtime is used to refer to periods when a system is unavailable.

Note: There are various Tools and Techniques for handling the *Load Balancing*, *Failover* situations and managing *High availability* of applications. However this is beyond the scope of this article.

As described above, the three key aspects have a significant impact on the overall smooth running of any business application. In the context of SAP MDM, this is achieved to some extent through setting up of distributed "Master" and "Slave" servers.

Typically, the "Slave" servers will have the same hardware specification as the "Master" server with its own set of master data objects residing on same or different database servers.

The other important terminologies to be understood are as follows:

iv. Benchmarking

In this case, substantial load is placed on the new hardware, software and the backend database. The performance of the system, business application and the user is monitored and recorded to be cross compared with other platforms. Thereby, enabling "benchmark" testing against various platforms, applications and scenarios.

SAP has provided standard application benchmarks and scalability tests on the following site:

www.sap.com/benchmark

v. SAPS

As per SAP –

The SAP Application Performance Standard (SAPS) is a hardware-independent unit that describes the performance of a system configuration in the SAP environment. It is derived from the Sales and Distribution (SD) Benchmark, where 100 SAPS is defined as the computing power to handle 2,000 fully business processed order line items per hour.

More information about SAPS: <http://www.sap.com/benchmark> -> SAP Benchmark Glossary -> SAPS

vi. Initial Sizing

This describes a typical platform independent hardware resource requirement for standard delivery of SAP application considering optimal system settings, standard business scenarios, etc.

vii. Expert Sizing

In this approach a detailed Sizing exercise is carried out on customer specific data and customized application to determine the hardware resource requirements, unlike the “Initial sizing” information provided for SAP standard application.

viii. Configuration and System Landscaping

SAP provides “best practices” that may be valid for specific combination of an operating system and database. Detailed information can be obtained from <http://service.sap.com/instguides> -> SAP NetWeaver.

What is Sizing?

As per SAP-

Sizing is determining the hardware requirements of an SAP application, such as the network bandwidth, physical memory, CPU processing power, and I/O capacity.

In case of SAP MDM application, this could be a very complex activity since it involves hardware requirements for the application considering the “Master” and “Slaves” repositories and where they may be installed.

Why is sizing important?

The most important reason for sizing is to ensure that there are no resource bottlenecks due to inadequate hardware configuration and the configuration of the system itself.

Today due to extensive use of IT, there are exacting demands on the standard software to provide excellent performance in terms of response time and data throughput. In the context of SAP MDM, acting as a master data hub, good response times are especially important when business applications depend on the real-time availability of cleansed master data.

Besides considering the SAP MDM application components, it's also important to consider the hardware requirements for the backend RDBMS.

As was mentioned in the previous article, the data modeling of the MDM repository must be very carefully developed. There must be a right balance between – Optimal Storage and Optimal Retrieval. Both these are exact opposites of each other. Since focus on any one of them has an adverse effect on the other. In which case, even with the best Sizing done, the performance may not be as expected.

As shown in the figure 1, the next key aspects to be considered are the factors that have impact on the performance namely;

- Repository data size
- Repository structure
- Concurrent connections
- Write back cache of DB hardware RAID.

There is substantial information the above mentioned factors in the SAP document – MDM 7.1 Sizing guide.pdf.

The key point to be noted is that the Data modeling of the Repository is one of the key factors that have an impact on the performance of the MDM application. Hence the “Data Profiling / Analysis / Discovery” is a very crucial activity prior to development phase (or working on the SAP MDM tool itself).

In the context of SAP NetWeaver MDM, the “Sizing” is to be carried out considering the three primary layers of the application (as seen in figure 2) namely; DBMS layer, MDM Server Layer and MDM Client component layer.

A. DBMS

As you are well aware, SAP MDM uses the DBMS as a container to store master data information including textual, images, PDF's, Video, Audio and other types of information needed to publish to multiple channels such as Portal, “relevant” OLTP and OLAP systems.

Typically, the master data records are comparatively smaller than the transaction records. However, the amount and type of information (as mentioned earlier) that defines a master object can be quite large. Hence a larger server configuration is recommended.

B. MDM Servers

The three hardware parameters that have great impact on the MDM sizing are:

- CPU
- Physical memory
- Local disk size

Typically, the MDM server utilizes RAM space for most of the processing. Hence more the RAM area available the better will be the processing and the performance of the MDM server.

As suggested by SAP –

The MDM Server is not certified for working in the memory swapping mode and, as such, an adequate amount of physical memory must be installed according to the most recent sizing guide.

Using CPUs that do not meet the sizing requirements will cause a performance bottleneck.

If the disk size is not sufficient, there will not be enough space for the creation of MDM accelerators and the repository loading will crash.

C. MDM Client components

Typically, the MDM client components will be installed on individual desktop / laptops depending on the role based requirements of the business user.

SAP has provided minimum and recommended hardware layout for the MDM client components in the MDM 7.1 Sizing guide.

Summary

In order to ensure optimal performance of the SAP MDM application, “Sizing” is one of the crucial aspects that need to be considered prior to “installation” of the application. The primary objective of “Sizing” is to ensure that there are no resource bottlenecks by the system.

Related Content

Kindly refer to the following SAP documents in details regarding Sizing.

1. “Good to Know” Topics related to Sizing and Performance.
(<http://service.sap.com/~sapidb/011000358700001191122007E>)
2. <https://www.sdn.sap.com/irj/sdn/nw-mdm>

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.