

SAP MDM Qualified Tables -a Deep Insight: Part - I



Applies to:

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

Summary

This article describes the basic information about qualified table concept of SAP MDM. It describes the actual internal structure of qualified table and how different sub-tables are linked together.

Author: Suresh Yadav

Company: Wipro Technologies

Created on: July 15, 2009

Author Bio



Suresh Yadav graduated in Electronics Engineering at the University of Pune, India. In 2006, he joined Wipro Technologies as a member of the SOA Competency Team. He has worked on SAP NetWeaver technology on topics such as Enterprise Portal, Knowledge Management, SAP Interactive Forms and SAP MDM. Since 2004 he has been working on the Java environment.

Table of Contents

Introduction	3
Use of Qualified Table	3
Inner structure of Qualified Table	5
One Qualified Look-Up record is used for storing qualifiers for all main table records? How	5
Qualified Table from Data Manager Perspective.....	6
Main Table View.....	6
Qualified Look-Up Table Values	6
Qualified Link Table View	7
Conclusion	7
Related Content.....	8
Disclaimer and Liability Notice.....	9

Introduction

Qualified Table – Did this term raised some questions in your mind? If yes, they shouldn't arise once you have gone through this article.

Qualified Table - Like flat table stores data in Rows and Columns then why it is a point of concern for everyone working in MDM workspace.

In the first part of this article I have tried to cover following points

- Use of Qualified Table
- Inner structure of a qualified table

Use of Qualified Table

Qualified table is used to store data that applies to all the records in the main table. If this data is stored in Main table this will increase the size of the repository by many times. Also this will result in duplication of Main table records.

To illustrate the above point lets consider an example. Suppose the repository store Products (e.g. Shirts) manufactured by an XYZ company. Some of the attributes of a product are Product-ID, Name, color, size, manufactured date, Price, etc. These attributes are mostly part of the Main table.

Product-ID	Name	Color	Size (inches)	Manufactured Date	Price (Rs)
P1	Name1	Light Blue	30	1/1/2009	1000

Fig: 1.0

Apart from these attributes which directly link to the product there are attributes whose value depend upon other attributes. For e.g. to increase sales manufacturers give offers in terms of lower price value against the ordered stock quantity. These varying prices are because of the promotional offers. Suppose we have the following data in hand.

Price: Rs.1000/- Quantity: 1
 Price: Rs.950/- Quantity: 10
 Price: Rs.900/- Quantity: 15

If this data is stored in Main table it will lead to duplication of Product data, as reflected from the below table.

Product-ID	Name	Color	Size (inches)	Manufactured Date	Quantity	Price (Rs)
P1	Name1	Light Blue	30	1/1/2009	1	1000
P1	Name1	Light Blue	30	1/1/2009	10	950
P1	Name1	Light Blue	30	1/1/2009	15	900

Fig: 1.1

Using qualified tables this data can be efficiently stored and also reused for other products.

In MDM world the dependent entities (in this case 'Price') are referred as Yes-Qualifiers and independent entities (in this case 'Quantity') are referred as No-Qualifiers.

Products Table - Main Table						
Product-ID	Name	Color	Size (inches)	Manufactured Date	Price (Qualified Look-Up Field)	
P1	Name1	Light Blue	30	1/1/2009	1;10;15	

Price Table - Qualified Table	
Quantity	Price
1	1000
10	950
15	900

This Field holds references to records of a Qualified Table

Fig - 1.2: Three records have been created in Qualified Table which refer product 'P1'.

Products Table - Main Table					
Product-ID	Name	Color	Size (inches)	Manufactured Date	Price (Qualified Look-Up Field)
P1	Name1	Light Blue	30	1/1/2009	1;10;15
P2	Name2	Navy Blue	30	1/1/2009	10;15

Price Table - Qualified Table	
Quantity	Price
10	850
15	800

This Field holds references to records of a Qualified Table

Fig - 1.3: Two records have been created in Qualified Table which refer product 'P2'.

The records created in Qualified Table are referred as **Links**. Each link record contains Yes and No qualifier field values. Thus we have avoided duplication and improved the performance.

Everything clear till this point? If yes, lets move on to next section. But I have a question, what is Qualified Look-Up Value? Referring to figure 1.2 and 1.3 the content of the qualified table 'Price' in all will be

Price Table - Qualified Table	
Quantity	Price
1	1000
10	950
15	900
10	850
15	800

For Product P1

For Product P2

The values (10 and 15) of Quantity field are repeated for storing price of product 'P2'. If we have 100 products with various discount schemes (on quantities 1/10/15) these values will be repeated 100 times which can be easily avoided. To avoid this we will make the Quantity field (No-Qualifier in Price Qualified Table) as flat look-up field and store these values in a separate flat look-up table. This flat look-up table is actually the **Qualified Look-Up** table.

Products Table - Main Table					
Product-ID	Name	Color	Size (inches)	Manufactured Date	Price (Qualified Look-Up Field)
P1	Name1	Light Blue	30	1/1/2009	1;10;15
P2	Name2	Navy Blue	30	1/1/2009	10;15

Quantities Table - Qualified LookUp Table	
RecordID	Quantity
QR1	1
QR2	10
QR3	15

Price Table - Qualified Link Table	
Quantity	Price
QR1	1000
QR2	950
QR3	900
QR2	850
QR3	800

Fig - 1.4

To sum up we have now

- Main Table – Products with qualified look-up field ‘Price’ that looks into Qualified Link Table ‘Price’,
- Qualified Link Table ‘Price’ with Quantity as flat look-up field that looks into Qualified Look-Up Table ‘Quantities’
- Records of Qualified Link Table ‘Price’ are referred as Links

Inner structure of Qualified Table

Qualified Table is always comprised of two tables, Qualified Look-Up table and Qualified Links table.

Qualified Look-Up table to store the No-Qualifiers and Qualified Links Table is used to store the links (records) comprised of Yes-Qualifier Field values (e.g. Price) against **foreign key of No Qualifier** values and **foreign key of Main Table**.

Products Table - Main Table						
RecordID	Product-ID	Name	Color	Size (inches)	Manufactured Date	Price (Qualified Look-Up Field)
PR1	P1	Name1	Light Blue	30	1/1/2009	1;10;15
PR2	P2	Name2	Navy Blue	30	1/1/2009	10;15

Price Table – Qualified Table comprised of Qualified Link and Look-Up table					
Quantities Table - Qualified LookUp Table		Price Table - Qualified Link Table			
RecordID	Quantity	RecordID	Products(Foreign Key)	Quantity(Foreign Key)	Price
QR1	1	PrR1	PR1	QR1	1000
QR2	10	PrR2	PR1	QR2	950
QR3	15	PrR3	PR1	QR3	900

Fig – 1.5

Note: The RecordID field in all the above tables cannot be viewed through the Client Application (MDM Data-Manager). This field value is internally generated when record is created in any table. When we view the content of a Qualified Table (against a Main Table record) in Data Manager we are only shown the No and Yes Qualifier fields. The Main Table record Id (Foreign Key) field is not shown to the user.

Referring to Fig – 1.5,

- PrR1, PrR2 and PrR3 are Qualified link Id
- QR1, QR2 and QR3 are Qualified Look-Up Id
- PR1 and PR2 are Main Table Record Id

One Qualified Look-Up record is used for storing qualifiers for all main table records? How

This statement can be easily understood if read from Right to left and stated as “Different Qualifier Records (Links) for one or more Main Table record can use one Qualified Look-Up record”.

In Fig 1.5 it's evident that Qualified Look-Up (Quantities Table) record ‘QR1’ is used in 3 records of Qualified Table (Prices) for storing quantity dependent price values of product ‘P1’.

Qualified Table from Data Manager Perspective

Using Data Manager we follow a sequence for reaching qualified link that is Main Table Record -> Qualified Look-Up Record -> Qualified Link Record.

Main Table View

Records							
	Product-ID	Name	Color	Size	Manufactured Date	Price	
	P1	Name1	Light Blue	30	07/01/2009	10; 15; 1	
	P2	Name2	Navy Blue	30	06/01/2009	10; 15	

Record Detail		Validations	Assignments	Workflows	Search Selections
Product-ID	P1				
Name	Name1				
Color	Light Blue				
Size	30				
Manufactured Date	07/01/2009				
Category					
		Masks			
		Price		1	
		[3 of 3]		10	
		<input checked="" type="checkbox"/> Filter		15	

Fig – 1.6: Products Table

Qualified Look-Up Table Values

The Qualified Look-Up table is embedded within the qualified table when observed from Data Manager. In our case if I open the Price Qualified table, its content will be the content of the Qualified Look-Up Table 'Quantities'. Adding a record in this table creates look-up records which are utilized in Qualified Links table.

Records	
	Quantity
	1
	10
	15

Record Detail		Validations	Assignments	Workflows
Quantity	1			
[Price]				

Fig – 1.7: Content of Price Table in Record Mode

Qualified Link Table View

This table is not explicitly available for viewing all its content from Data Manager. The content of this table can only be viewed against a Main table record. As illustrated in Fig -1.8 a link record is created in Qualified Link table by selecting a look-up record in left pane and adding it to the right pane. Hence when a link is created its No-Qualifier field is automatically populated whereas the Yes-qualifier fields values can be entered in the below pane. Internally the foreign key of the No-Qualifier record is added to the link and also the foreign key of the Main table record is also added.

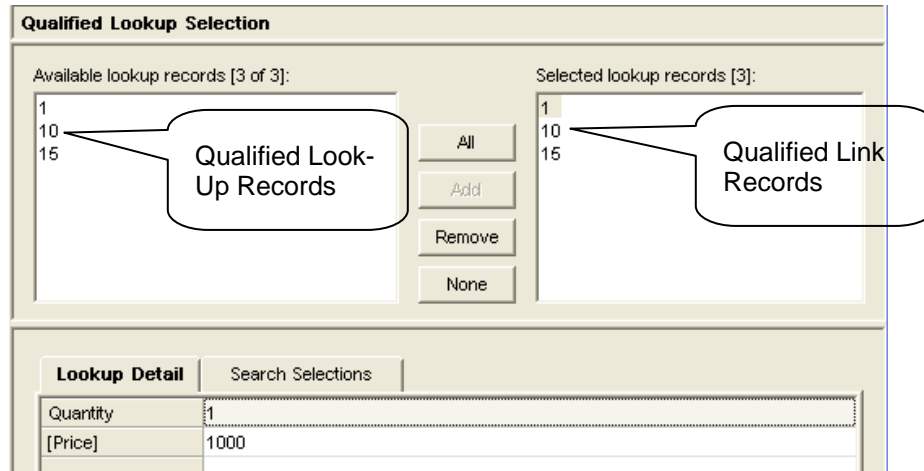


Fig – 1.8: Content of Price Table when viewed against a main table record

Conclusion

An MDM Qualified table is composed of two flat tables, Qualified Look-Up table and Qualified Link table. The No-qualifiers are stored in Qualified Look-Up table; Yes-qualifiers along with references to No-qualifiers and Main table record-Id are stored in Qualified Link table.

Related Content

<https://websmp210.sap-ag.de/installMDM71-> SAP MDM Console Reference Guide

<https://websmp210.sap-ag.de/installMDM71-> SAP MDM Data Manger Reference Guide

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

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.