

# Integrating Sales Conditions Data with Sales Item level Data in SAP BW



## Applies to:

SAP BIW 3.X, SAP NetWeaver 7.0.

## Summary

While implementing a business data warehouse for Sales data from SAP R3, it is essential to integrate the various conditions data (order and billing conditions) to the item level data. There are separate extractors available to extract the order/billing conditions data and the item level data to BW. This document illustrates how to design the BW data model for efficient reporting of the same.

**Author:** Yogesh Kulkarni

**Company:** Infosys Technologies Ltd.

**Created on:** 7 July 2010

## Author Bio



Yogesh Kulkarni is presently working with Infosys Technologies Ltd. He has worked extensively on SAP BIW 3.5, SAP NetWeaver BI 2004s with significant exposure to BOBJ (Universe Designer, WebI, Live Office, and Xcelsius).

## Table of Contents

What is Conditions Data in Sales Data Model? .....	3
How to Model? .....	3
Key Figure Based Modeling.....	4
Advantages .....	5
Disadvantages .....	5
Characteristics Based Modeling .....	5
Related Content .....	5
Disclaimer and Liability Notice.....	6

## What is Conditions Data in Sales Data Model?

The pricing conditions are various elements of the value of the product/commodity. There can be standard pricing elements (conditions) like *VAT*, *Freight* etc in addition to the custom pricing conditions implemented for specific business processes.

In SAP R3, these conditions are stored in separate tables like KONV, KOMV (other than the standard transactional data tables like VBRK, VBRP). There are standard business contents extractors available to extract such data into SAP BW. These extractors are namely 2LIS\_11\_VAKON (Orders Conditions Data) and 2LIS\_13\_VDKON (Billing Conditions Data).

## How to Model?

As the transactional data like orders\billing and conditions data are extracted separately into SAP BW, it becomes essential to integrate them into SAP BW Data Model.

As a part of Enterprise Data Warehouse, it is advised to extract the sales order items, billing items, sales order conditions, and billing condition data to separate DSOs. This way we can make sure that we have the data from Source System (SAP R3 in this case) AS IS in our EDW layer.

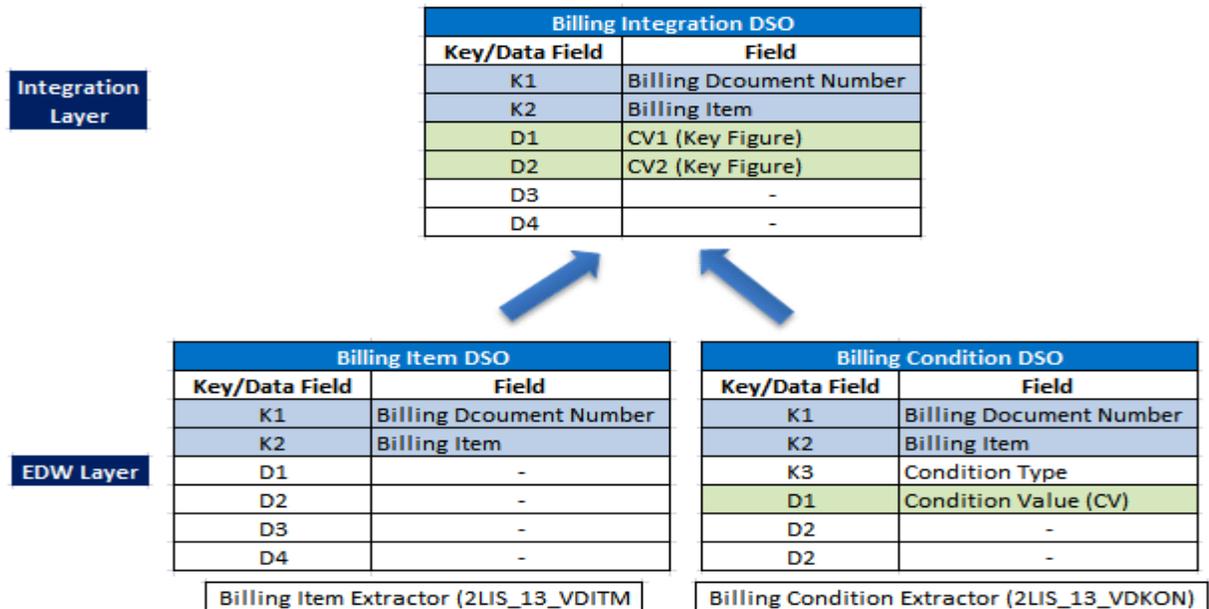
There could be two ways to develop integration model in SAP BW.

1. Key Figure Based Modeling – In this Model, we add the separate key figures for each and every condition type into the Integration DSO. This modeling can also be called as *Column based modeling*.
2. Characteristic Based Modeling – In this model, we introduce the Condition Type as a key field in the Integration DSO. This modeling can also be called as *Row Based Modeling*.

In the subsequent sections, we will discuss both type of modeling in detail with its advantages and disadvantages.

## Key Figure Based Modeling

In this Model, we add the separate key figures for each and every condition type into the Integration DSO. This modeling can also be called as *Column based modeling*. Following diagram illustrates the dataflow for the same.



1. Billing Item data is extracted to a *Billing Item DSO* using 2LIS\_13\_VDITM and Billing Conditions data is extracted to a separate *Billing Conditions DSO* using 2LIS\_13\_VDKON.
2. The keys of *Billing Item DSO* are mentioned as Billing Document Number, Billing Item. (Please refer the above diagram).
3. The keys of *Billing Conditions DSO* are mentioned as Billing Document Number, Billing Item and Condition Type including some others. (Please refer the above diagram).
4. The *Billing Integration DSO* will have keys similar to *Billing Item DSO* that is Billing Document Number and Billing Item.
5. The *Billing Integration DSO* will be added with separate key figures as CV1, CV2... etc for each billing condition type (Key Figures based modeling).
6. While updating these added key figure values in the transformation (from *Billing Item DSO* to *Billing Integration DSO*), we will look up the *Billing Condition DSO* based on key fields Billing Document Number and Billing Item (which are similar in both the DSOs) and the Condition Type = *respective Condition Type*. The pseudo code would be as follows...

Read *Billing Condition DSO* where

Billing Document Number (Item DSO) = Billing Document Number (Condition DSO) and

Billing Item (Item DSO) = Billing Item (Condition DSO) and

Condition Type = '*Respective condition type*'. (Hard Coding).

Assignment → CV1 (Key Figure for first Condition type in Integration DSO) = Condition Value (Condition DSO)

7. Similarly we can update all other added key figures by hard coding the respective *condition type* in their look up logic.
8. Finally in the *Billing Integration DSO*, we will have the integrated record for each Billing Document having the various condition values in their respective key figures.

## Advantages

1. The data redundancy will be reduced as there will be only single record in the Integration DSO for each Billing Document and Item combination. Note that Condition DSO has multiple records (one each for each pricing condition) for every Billing Document & Item combination.
2. Flexible to implement. Depending upon the number of condition types for which we need the data, we can add those many numbers of key figures in the Integration DSO.
3. The EDW layer will always have the data for all the condition types. Hence in case we need additional condition types for reporting, there is no need to go back to source system again.

## Disadvantages

1. The little bit of ABAP is involved in implementing it. We cannot avoid the same as the granularity of the data in Billing Items and Billing Conditions is different. Hence we cannot integrate these data directly.

## Characteristics Based Modeling

In this model, we introduce the Condition Type as a key field in the Integration DSO. As for given Billing Document and Item, for every condition type, there would be separate row (record) in the DSO; hence this modeling can also be called as Row Based Modeling.

This way, the granularity of the Conditions DSO and the Integration DSO remains the same and each Billing Document and Item combination has multiple records based on the various condition types. Also for a given Billing Document and Item combination, you will never get all the condition values in single record. As the data redundancy and volume of the data increases significantly, this type of modeling is not advised.

## Related Content

[http://help.sap.com/erp2005\\_ehp\\_04/helpdata/EN/ca/1885403e1b6e1de10000000a1550b0/frameset.htm](http://help.sap.com/erp2005_ehp_04/helpdata/EN/ca/1885403e1b6e1de10000000a1550b0/frameset.htm)

<http://forums.sdn.sap.com/thread.jspa?threadID=120492>

[http://help.sap.com/erp2005\\_ehp\\_04/helpdata/EN/7f/0f8c4037fba62be10000000a1550b0/frameset.htm](http://help.sap.com/erp2005_ehp_04/helpdata/EN/7f/0f8c4037fba62be10000000a1550b0/frameset.htm)

## **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.