Data Model Changes Regarding SD Index Tables

Document Version	Status	Date
1.0	final	October 20, 2015

Table of Content

1 Data Model Changes

1.1 General Approach

1.1.1 Motivation

There are a number of database tables which, as a standard deployment, contain only redundant data that originates from the corresponding document tables. These tables were originally introduced to accelerate lookup processes for documents that are based on attributes distributed across various document tables (for example, order header, order line item, and order partner assignment).

In S4/HANA, the database enables you to search for documents that refer to attributes from multiple tables while efficiently handling distributed selectiveness. Therefore, the decision was made to remove these redundant tables from the persistency layer, and replace it with views populated by the very same information.

1.1.2 Related Dictionary Object

Table	Short Description
VAKPA	Sales Index: Orders by Partner Function
VAPMA	Sales Index: Order Items by Material
VLKPA	SD Index: Deliveries by Partner Functions
VLPMA	SD Index: Delivery Items by Material
VRKPA	Sales Index: Bills by Partner Functions
VRPMA	SD Index: Billing Items per Material

S4/HANA provides a set of corresponding views. The following table shows which view (and DDL source) corresponds to which table. The views listed are CDS views, which can be accessed directly in ABAP.

Table	CDS view	DDL source
VAKPA	V_VAKPA_CDS	V_VAKPA
VAPMA	V_VAPMA_CDS	V_VAPMA
VLKPA	V_VLKPA_CDS	V_VLKPA
VLPMA	V_VLPMA_CDS	V_VLPMA
VRKPA	V_VRKPA_CDS	V_VRKPA
VRPMA	V_VRPMA_CDS	V_VRPMA

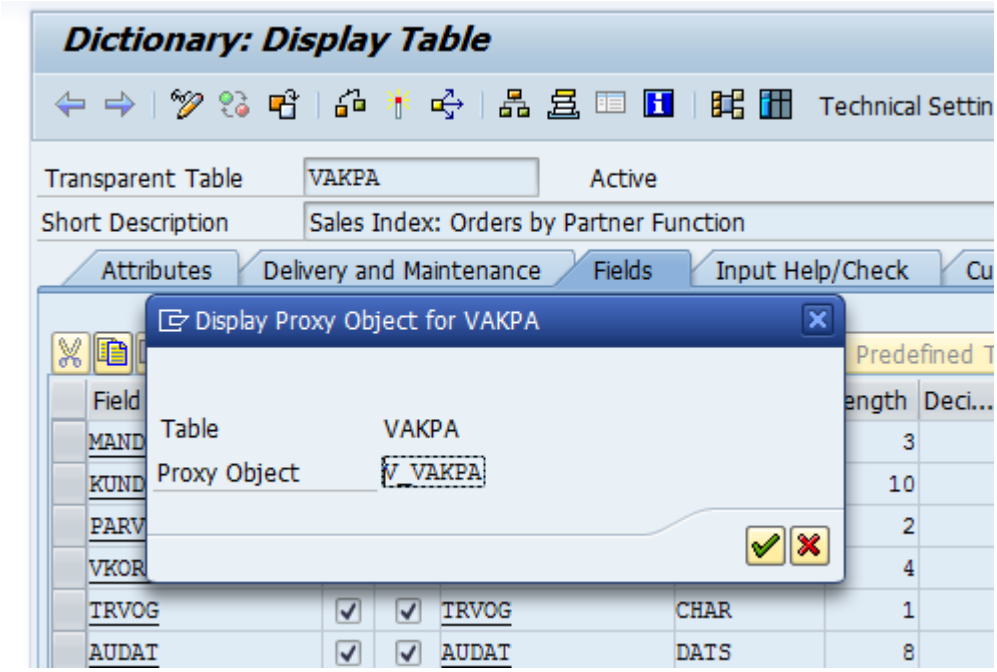
The original tables will remain in the system and can, for example, be used for typing but will not be populated with data. Inserts or any other modifying operations are no longer permitted in these tables.

1.2 Compatibility Mode

1.2.1 Standard Deployment

In SD, we explicitly decided to replace all access processes to the original index tables with access processes to the new views. These six tables support compatibility mode.. Each original index table is equipped with a proxy object linking the CDS view to the index table.

Example for VAKPA:



In cases where the original index table is not enhanced, all select statements will automatically function just as as before, meaning there is no need for any adoption.

However, since the persistency layer becomes obsolete, manipulating statements that access these tables are not permitted.

1.2.2 How to Handle Enhancements

1.2.2.1 Defining Extension Views

Extension views can be used to enhance the compatibility views in order to populate with attributes. Before this optimization, these attributes were stored in the material index or partner index tables.

Since there is no longer a persistency layer, the first step towards moving the enhancement attributes to extend views is to decide to which of the actual database tables the compatibility views refer to the attributes that are to be relocated. This relocation requires a custom build migration report that has to read the original index tables before the compatibility views are introduced. It also has to move the data to the underlying sales document tables, which now carry all attributes that make up the compatibility views.

The extension view must contain the very same attributes as the extension of the original index-table. Please be aware that the sequence of the attributes must be the same as it is in the original tables (more precisely: in the runtime object). If the extension views don't match any of these pre-requisites, they cannot be activated because the runtime objects for both the actual index table and the compatibility view are not identical.

The following examples show how extension views for the index tables can be defined. There is a specific example for material index table VAPMA and partner index table VAKPA.

These examples are based on a DDIC object in the local namespace ‘Z’, but contain the very same information as the actual tables.

For more information about extension views, see::

http://help.sap.com/abapdocu_740/en/index.htm?file=abencds_f1_extend_view.htm

1.2.2.2 Example: How toxtend VAPMA

1.2.2.2.1 CDS View Definition

The following data definition represents the compatibility view for VAPMA:

```
@AbapCatalog.sqlViewName: 'Z_V_VAPMA_CDS'
@ClientDependent: true
@AbapCatalog.compiler.compareFilter: true
@EndUserText.label: 'compatiblity view for VAPMA'
define view Z_V_VAPMA as

select

    from vbak as vbak

    join vbap as vbap on

         vbap.vbeln = vbak.vbeln

    left outer join vbpa as vbpa on

         vbpa.vbeln = vbap.vbeln
     and vbpa.parvw = 'AG'
     and vbpa.posnr = '000000'

{
    vbak.mandt  as mandt,
    vbap.matnr  as  matnr,
    vbak.vkorg  as  vkorg,
    vbak.trvog  as  trvog,
    vbak.audat  as  audat,
    vbak.vtweg  as  vtweg,
    vbak.spart  as  spart,
    vbak.auart  as  auart,
    vbpa.kunnr  as  kunnr,
    vbak.vkbur  as  vkbur,
    vbak.vkgrp  as  vkgrp,
    vbak.bstnk  as  bstnk,
    vbak.ernam  as  ernam,
    vbak.vbeln  as  vbeln,
    vbap.posnr  as  posnr,
    vbap.werks  as  werks,
    vbpa.adrnr  as  adrnr,

     case vbak.vbtyp
        when 'A' then vbak.angdt
        when 'B' then vbak.angdt
     else vbak.guebg
     end  as datab,

     case vbak.vbtyp
        when 'A' then
            case vbak.bnddt when '00000000' then '99991231'
            else vbak.bnddt
            end
        when 'B' then
            case vbak.bnddt when '00000000' then '99991231'
```

```
            else vbak.bnddt
            end
        else
            case vbak.gueen when '00000000' then '99991231'
            else vbak.gueen
            end
    end as datbi
}
```

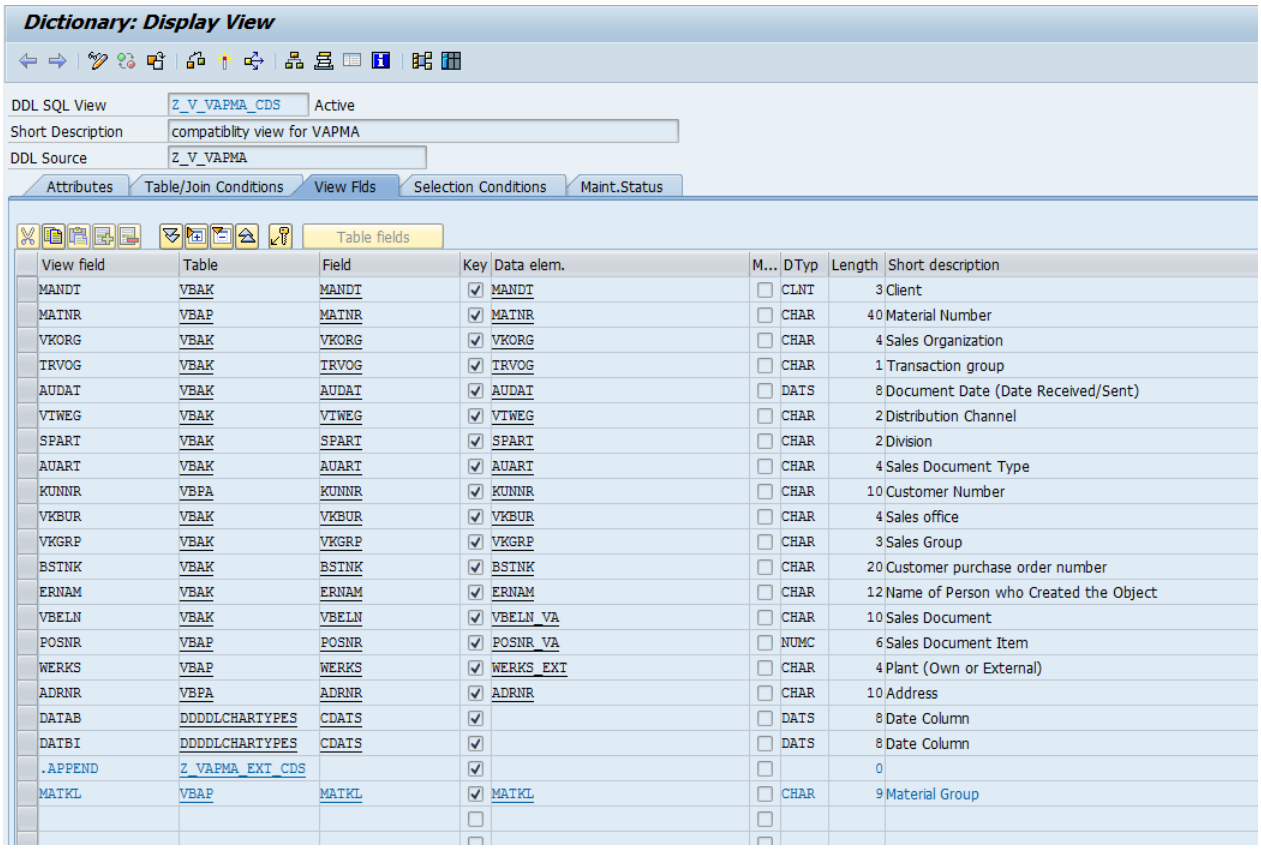
1.2.2.2.2 Extension View

Example:

The original VAPMA table was extended by MATKL (from table VBAP), therefore MATKL must also be added to the compatibility view as an extension view:

```
@AbapCatalog.sqlViewAppendName: 'Z_VAPMA_EXT_CDS'
@EndUserText.label: 'extension view to Z_V_VAPMA'
extend view Z_V_VAPMA with Z_V_VAPMA_EXT {
    vbap.matkl as matkl
}
```

After activating the extension, an append view is automatically generated for view Z_V_VAPMA_CDS:



View field	Table	Field	Key	Data elem.	M...	DTyp	Length	Short description
MANDT	VBAK	MANDT	✓	MANDT		CLNT	3	Client
MATNR	VBAP	MATNR	✓	MATNR		CHAR	40	Material Number
VKORG	VBAK	VKORG	✓	VKORG		CHAR	4	Sales Organization
TRVOG	VBAK	TRVOG	✓	TRVOG		CHAR	1	Transaction group
AUDAT	VBAK	AUDAT	✓	AUDAT		DATS	8	Document Date (Date Received/Sent)
VTWEG	VBAK	VTWEG	✓	VTWEG		CHAR	2	Distribution Channel
SPART	VBAK	SPART	✓	SPART		CHAR	2	Division
AUART	VBAK	AUART	✓	AUART		CHAR	4	Sales Document Type
KUNNR	VBPA	KUNNR	✓	KUNNR		CHAR	10	Customer Number
VKBUR	VBAK	VKBUR	✓	VKBUR		CHAR	4	Sales office
VKGRP	VBAK	VKGRP	✓	VKGRP		CHAR	3	Sales Group
BSTNK	VBAK	BSTNK	✓	BSTNK		CHAR	20	Customer purchase order number
ERNAM	VBAK	ERNAM	✓	ERNAM		CHAR	12	Name of Person who Created the Object
VBELN	VBAK	VBELN	✓	VBELN_VA		CHAR	10	Sales Document
POSNR	VBAP	POSNR	✓	POSNR_VA		NUMC	6	Sales Document Item
WERKS	VBAP	WERKS	✓	WERKS_EXT		CHAR	4	Plant (Own or External)
ADRNR	VBPA	ADRNR	✓	ADRNR		CHAR	10	Address
DATAB	DDDDLCHARTYPES	CDATS	✓			DATS	8	Date Column
DATBI	DDDDLCHARTYPES	CDATS	✓			DATS	8	Date Column
.APPEND	Z_VAPMA_EXT_CDS		✓				0	
MATKL	VBAP	MATKL	✓	MATKL		CHAR	9	Material Group

In the runtime object of the view the attribute MATKL is added to the end, similar to the original index table VAPMA:

Display: Active runtime object for Z_V_VAPMA_CDS

Database View: Z_V_VAPMA_CDS
Runtime Object Time Stamp: 10.08.2015 15:39:07
ABAP Time Stamp: 10.08.2015 15:39:08
DYNP Time Stamp: 10.08.2015 15:39:08

Header of active runtime object

Obj...	Dat...	No...	Tabl...	No...	Ke...	Po...	Ali...	Buf...	Nu...	Le...	Flag 1	P...	DB	Flag 2	Flag 3	Flag 4	Flag 5	Flag 6	Poi...	Uni...	UUID	Leaf...	Con
J	T	20	336	20	336	1	2		0	0	00010110		B	00000000	00000000	00000000	00000000	00000000	0	2	F8B2C255D8D995...	0	000

Fields of active runtime object

Field Name	Fiel...	De...	Data...	ABA...	DB len...	De...	Field...	Ou...	Fiel...	AB...	AB...	Dic...	Flag 1	Flag 2	Flag 3	Flag 4	Data element	Pre...	R...	R...	Conv...	SET/...	Check t
MANDT	1	0	CLNT	6	6	0	0	3	152	0	C	0	00000001	00001000	00000010	00000000	MANDT	0					T000
MATNR	2	0	CHAR	80	80	0	6	40	40	0	C	0	00100001	00001110	00000010	00000000	MATNR	0			MAT...	MAT	MARA
VKORG	3	0	CHAR	8	8	0	86	4	40	0	C	0	00100001	00001100	00000010	00000000	VKORG	0				VKO	TVKO
TRVOG	4	0	CHAR	2	2	0	94	1	40	0	C	0	00010001	00000100	00000010	00000000	TRVOG	0				VTV	
AUDAT	5	0	DATS	16	16	0	96	10	156	1	D	0	00100001	00000000	00000010	00000000	AUDAT	0					
VTWEG	6	0	CHAR	4	4	0	112	2	40	0	C	0	00100001	00001100	00000010	00000000	VTWEG	0				VTW	TVKOV
SPART	7	0	CHAR	4	4	0	116	2	40	0	C	0	00100001	00001100	00000010	00000000	SPART	0				SPA	TVTA
AUART	8	0	CHAR	8	8	0	120	4	40	0	C	0	00000001	00001110	00000010	00000000	AUART	0			AUA...	AAT	TVAK
KUNNR	9	0	CHAR	20	20	0	128	10	40	0	C	0	00100001	00001110	00000010	00000000	KUNNR	0			ALP...	KUN	KNA1
VKBUR	10	0	CHAR	8	8	0	148	4	40	0	C	0	00100001	00001100	00000010	00000000	VKBUR	0				VKB	TVKBZ
VKGRP	11	0	CHAR	6	6	0	156	3	40	0	C	0	00100001	00001100	00000010	00000000	VKGRP	0				VKG	TVBVK
BSTNK	12	0	CHAR	40	40	0	162	20	40	0	C	0	00000101	00000000	00000000	00000000	BSTNK	0					
ERNAM	13	0	CHAR	24	24	0	202	12	40	0	C	0	00000001	00000000	00000000	00000000	ERNAM	0					
VBELN	14	0	CHAR	20	20	0	226	10	40	0	C	0	00000001	00000110	00000011	00000000	VBELN_VA	0			ALP...	AUN	
POSNR	15	0	NUMC	12	12	0	246	6	176	6	N	0	00000001	00000100	00000011	00000000	POSNR_VA	0				APO	
WERKS	16	0	CHAR	8	8	0	258	4	40	0	C	0	00100001	00000100	00000010	00000000	WERKS_EXT	0				WRK	
ADRNR	17	0	CHAR	20	20	0	266	10	40	0	C	0	00000001	00000010	00000000	00000000	ADRNR	0			ALP...		
DATAB	18	0	DATS	16	16	0	286	10	156	1	D	0	00000001	00000000	00000010	00000000		0					
DATBI	19	0	DATS	16	16	0	302	10	156	1	D	0	00000001	00000000	00000010	00000000		0					
MATKL	20	0	CHAR	18	18	0	318	9	40	0	C	0	00100001	00001100	00000010	00000000	MATKL	0				MKL	T023

1.2.2.3 Example: How to Extend VAKPA

1.2.2.3.1 Specifics for partner views

Extending VAPMA is straightforward, since the compatibility view for VAPMA comprises only one data definition (one CDS view).

This is different for the partner-index compatibility views. VAKPA is used as an example – VRKPA, and VLKPA follow the same approach.

All partner-index views must comprise four different partner roles:

- Customer
- Supplier
- Employee
- Contact person
- Unloading point

For each of these roles (referring to an entry in table TPAR), a separate, distinct view is defined. This view is specific to the partner role. This is necessary because ,depending on the role, different attributes from the VBPA table are populated to the attribute KUNDE:

- Customer → vbpa.kunnr becomes KUNDE
- Supplier → vbpa.lifnr becomes KUNDE
- Employee → vbpa.pernr becomes KUNDE
- Contact person → vbpa.parnr becomes KUNDE
- Unloading point → vbpa.ablad becomes KUNDE

Each of these originating attributes is highly significant and must be used efficiently to scan VBPA when searching for the corresponding rows. In order to ensure this, the assignment to the target attribute KUNDE must be unconditionally, that means using a case expression in the attribute list, as shown the following code snippet, is not possible:

```
case tpar.nrart
   when 'KU' then cast(vbpa.kunnr as abap.char (  10 ))
   when 'LI' then cast(vbpa.lifnr as abap.char (  10 ))
   when 'AP' then cast(vbpa.parnr as abap.char (  10 ))
   when 'PE' then cast(vbpa.pernr as abap.numc (  10 ))
 else cast(vbpa.ablad as abap.char (  10 ))

 end  as kunde ,
```

Any access to the view containing

```
where kunde ='<xyz>‘
```

would lead to an inefficient scanning of VBPA in terms of generating a sub-result set that contains all values for KUNDE assigned from the multiple source attributes. This can easily lead to millions and millions of entries. Based on this internal sub-result set, the actual look-up for the requested value would be processed.

Instead, each role-specific view only contains an unconditional assignment (for partner role ‘customer’, for example):

```
vbpa.kunnr      as kunde,
```

The following chapters describe how the compatibility views for partner index tables are set up.

1.2.2.3.2 CDS View definition

The CDS view is a two-layer definition comprising 5 views (each for every partner role) on the lower level and the actual compatibility view for VAKPA on the top-level.

1.2.2.3.2.1 Top-level view representing VAKPA

The following data definition represents the compatibility view for VAPMA:

```
@AbapCatalog.sqlViewName: 'V_VAKPA_CDS'

// Attribute KUNDE is populated from different source-attributes based on the value stored
in tpar.nrat, which is derived from parvw.
//
// UNION all for separate selects differentially on tpar.nrart with the specifically
populating attribute KUNDE shows
// better performance than populating KUNDE via the case expression in the attribute list.
//
```

```
// This is because "transitivity"  , which states the source-attributes that are populated
to KUNDE, is not taken into account
// when choosing the optimal access path with a given when condition on kunde, which would
literally lead to retrieving column kunde for all
// rows as an intermediate result set and subsequently filtering on kunde. Especially
single column indices on the source attributes are not used.
//
// Using UNION all instead represents choosing specific and optimal query plans, and also
utilizing single column indices, which outperform the case version
// by a factor of up to 50.
//
// This approach also holds true for:
//    - V_VLKPA_CDS
//    - V_VRKPA_CDS

define view v_vakpa as

select from V_VAKPA_KU
{
*
}

union all

select from V_VAKPA_LI
{
*
}

union all

select from V_VAKPA_AP
{
*
}

union all

select from V_VAKPA_PE
{
*
}

union all

select from V_VAKPA_AB
{
*
}
```

The description of how to extend this type of partner-index compatibility views uses a simplified version of the original views that only cover the partner roles ‘customer’ and ‘supplier’. The other three roles follow the same approach.

Example for top-view:

```
@AbapCatalog.sqlViewName: 'Z_V_VAKPA_CDS'
@ClientDependent: true
@AbapCatalog.compiler.compareFilter: true
@EndUserText.label: 'compatiblity view for VAKPA'
define view Z_V_VAKPA as

select from Z_V_VAKPA_KU
{
*
}

union all

select from Z_V_VAKPA_LI
{
*
}
```

Please notice that the attribute lists representing the attributes retrieved from the role-specific views are generically declared using ‘*’. This is crucial for allowing this view to be extended.The following chapter provides more details about this.

1.2.2.3.2.2 Lower-Level Views for Customer and Supplier

The following data definition represents the subview for customers:

```
@AbapCatalog.sqlViewName: 'Z_V_VAKPA_KU_CDS'
@ClientDependent: true
@AbapCatalog.compiler.compareFilter: true
@EndUserText.label: 'sub-view for customer'
define view Z_V_VAKPA_KU as

select distinct

   from vbak as vbak

   join vbpa as vbpa on

        vbpa.vbeln  =  vbak.vbeln

   join vbpa as vbpa_ag on

        vbpa_ag.vbeln   =  vbak.vbeln
    and vbpa_ag.parvw   =  'AG'
    and vbpa_ag.posnr   =  '000000'

   join tpar as tpar on

        tpar.parvw  = vbpa.parvw
    and tpar.nrart = 'KU'

{
        vbak.mandt      as mandt,
        vbpa.kunnr      as kunde,
```

```
        vbpa.parvw      as parvw,
        vbak.vkorg      as vkorg,
        vbak.trvog      as trvog,
        vbak.audat      as audat,
        vbak.vkbur      as vkbur,
        vbak.vkgrp      as vkgrp,
        vbak.vtweg      as vtweg,
        vbak.spart      as spart,
        vbak.auart      as auart,
        vbak.bstnk      as bstnk,
        vbpa_ag.kunnr   as kunnr,
        vbak.ernam      as ernam,
        vbak.vbeln      as vbeln,
        vbpa_ag.adrnr   as adrnr_ag,
        vbpa.adrnr      as adrnr,

         case vbak.vbtyp
            when 'A' then vbak.angdt
            when 'B' then vbak.angdt
          else vbak.guebg
          end  as datab ,

         case vbak.vbtyp
            when 'A' then
                case vbak.bnddt when '00000000' then '99991231'
                else vbak.bnddt
                end
            when 'B' then
                case vbak.bnddt when '00000000' then '99991231'
                else vbak.bnddt
                end
            else
                case vbak.gueen when '00000000' then '99991231'
                else vbak.gueen
                end
          end  as datbi
}
```

Please note that the parts highlighted in yellow make this view specific to the partner role ‘customer’

The following data definition represents the subview for suppliers:

```
@AbapCatalog.sqlViewName: 'Z_V_VAKPA_LI_CDS'
@ClientDependent: true
@AbapCatalog.compiler.compareFilter: true
@EndUserText.label: 'sub-view for supplier'
define view Z_V_VAKPA_LI as

select distinct

   from vbak as vbak

   join vbpa as vbpa on

        vbpa.vbeln  =  vbak.vbeln

   join vbpa as vbpa_ag on

        vbpa_ag.vbeln   =  vbak.vbeln
    and vbpa_ag.parvw   =  'AG'
    and vbpa_ag.posnr   =  '000000'

   join tpar as tpar on

        tpar.parvw  = vbpa.parvw
    and tpar.nrart = 'LI'

{
        vbak.mandt      as mandt,
        vbpa.lifnr      as kunde,
        vbpa.parvw      as parvw,
        vbak.vkorg      as vkorg,
        vbak.trvog      as trvog,
        vbak.audat      as audat,
        vbak.vkbur      as vkbur,
        vbak.vkgrp      as vkgrp,
        vbak.vtweg      as vtweg,
        vbak.spart      as spart,
        vbak.auart      as auart,
        vbak.bstnk      as bstnk,
        vbpa_ag.kunnr   as kunnr,
        vbak.ernam      as ernam,
        vbak.vbeln      as vbeln,
        vbpa_ag.adrnr   as adrnr_ag,
        vbpa.adrnr      as adrnr,

         case vbak.vbtyp
            when 'A' then vbak.angdt
            when 'B' then vbak.angdt
          else vbak.guebg
          end  as datab ,

         case vbak.vbtyp
            when 'A' then
                case vbak.bnddt when '00000000' then '99991231'
                else vbak.bnddt
                end
            when 'B' then
                case vbak.bnddt when '00000000' then '99991231'
```

```
                else vbak.bnddt
                end
            else
                case vbak.gueen when '00000000' then '99991231'
                else vbak.gueen
                end
        end  as datbi
}
```

Please note that the parts highlighted in yellow make this view specific to the partner role ‘supplier’

1.2.2.3.3 Extension View

Extension views containing the set operation UNION (ALL) cannot be declared explicitly on the top-level view, because it’s not possible to declare which attributes should belong to which UNION branch. Instead, the extension must be declared on the lower-level views, which will be inherited by the top-level view, because the attribute list from the lower-level views is generically populated using ‘*’.

Attention:
All extensions to the lower-level views must be identical in terms of sequence and type of the extending attributes!

Example 1:

For all partner roles, the attribute LAND1 of table VBPA should be added to the compatibility view.

In order to achieve this, two extension views must be declared: one for customer and another one for suppliers.

Extension view for customers:

```
@AbapCatalog.sqlViewAppendName: 'Z_VKP_KU_EXT_CDS'
@EndUserText.label: 'extend view for sub-view customer'
extend view Z_V_VAKPA_KU with Z_V_VAKPA_KU_EXT {
    vbpa.land1 as land1
}
```

After activating the extension, an append view is automatically generated for view Z_V_VAKPA_KU_CDS:

DDL SQL View: Z_V_VAKPA_KU_CDS Active
Short Description: sub-view for customer
DDL Source: Z_V_VAKPA_KU

Attributes | Table/Join Conditions | View Flds | Selection Conditions | Maint.Status

Table fields

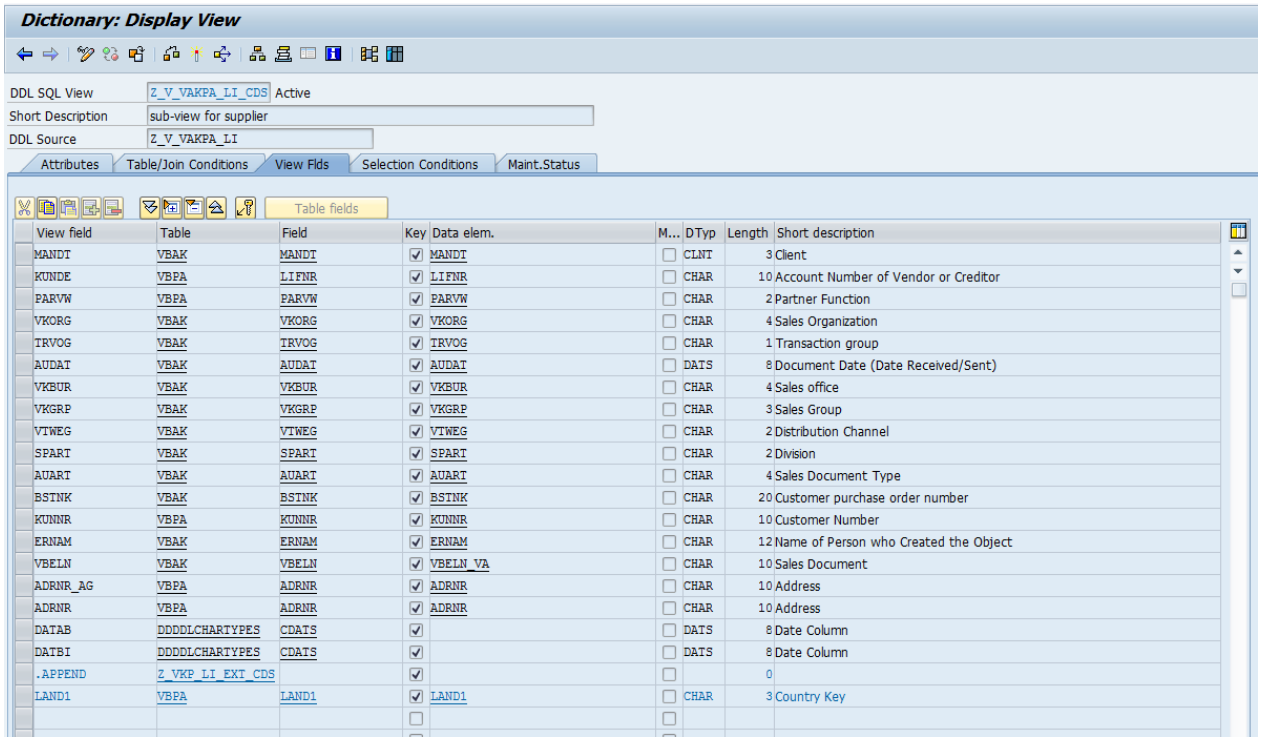
View field	Table	Field	Key	Data elem.	M...	DTyp	Length	Short description
MANDT	VBAK	MANDT	☑	MANDT	☐	CLNT	3	Client
KUNDE	VBPA	KUNNR	☑	KUNNR	☐	CHAR	10	Customer Number
PARVW	VBPA	PARVW	☑	PARVW	☐	CHAR	2	Partner Function
VKORG	VBAK	VKORG	☑	VKORG	☐	CHAR	4	Sales Organization
TRVOG	VBAK	TRVOG	☑	TRVOG	☐	CHAR	1	Transaction group
AUDAT	VBAK	AUDAT	☑	AUDAT	☐	DATS	8	Document Date (Date Received/Sent)
VKBUR	VBAK	VKBUR	☑	VKBUR	☐	CHAR	4	Sales office
VKGRP	VBAK	VKGRP	☑	VKGRP	☐	CHAR	3	Sales Group
VTWEG	VBAK	VTWEG	☑	VTWEG	☐	CHAR	2	Distribution Channel
SPART	VBAK	SPART	☑	SPART	☐	CHAR	2	Division
AUART	VBAK	AUART	☑	AUART	☐	CHAR	4	Sales Document Type
BSTNK	VBAK	BSTNK	☑	BSTNK	☐	CHAR	20	Customer purchase order number
KUNNR	VBPA	KUNNR	☑	KUNNR	☐	CHAR	10	Customer Number
ERNAM	VBAK	ERNAM	☑	ERNAM	☐	CHAR	12	Name of Person who Created the Object
VBELN	VBAK	VBELN	☑	VBELN_VA	☐	CHAR	10	Sales Document
ADRNR_AG	VBPA	ADRNR	☑	ADRNR	☐	CHAR	10	Address
ADRNR	VBPA	ADRNR	☑	ADRNR	☐	CHAR	10	Address
DATAB	DDDDLCHARTYPES	CDATS	☑		☐	DATS	8	Date Column
DATBI	DDDDLCHARTYPES	CDATS	☑		☐	DATS	8	Date Column
.APPEND	Z_VKP_KU_EXT_CDS		☑		☐		0	
LAND1	VBPA	LAND1	☑	LAND1	☐	CHAR	3	Country Key
			☐		☐			
			☐		☐			

Extension view for suppliers:

```
@AbapCatalog.sqlViewAppendName: 'Z_VKP_LI_EXT_CDS'
@EndUserText.label: 'extend view for sub-view customer'
extend view Z_V_VAKPA_LI with Z_V_VAKPA_LI_EXT {
    vbpa.land1 as land1

}
```

After activating the extension, an append view is automatically generated for view Z_V_VAKPA_LI_CDS



View field	Table	Field	Key	Data elem.	M...	DTyp	Length	Short description
MANDT	VBAK	MANDT	☑	MANDT	☐	CLNT	3	Client
KUNDE	VBPA	LIFNR	☑	LIFNR	☐	CHAR	10	Account Number of Vendor or Creditor
PARVW	VBPA	PARVW	☑	PARVW	☐	CHAR	2	Partner Function
VKORG	VBAK	VKORG	☑	VKORG	☐	CHAR	4	Sales Organization
TRVOG	VBAK	TRVOG	☑	TRVOG	☐	CHAR	1	Transaction group
AUDAT	VBAK	AUDAT	☑	AUDAT	☐	DATS	8	Document Date (Date Received/Sent)
VKBUR	VBAK	VKBUR	☑	VKBUR	☐	CHAR	4	Sales office
VKGRP	VBAK	VKGRP	☑	VKGRP	☐	CHAR	3	Sales Group
VTWEG	VBAK	VTWEG	☑	VTWEG	☐	CHAR	2	Distribution Channel
SPART	VBAK	SPART	☑	SPART	☐	CHAR	2	Division
AUART	VBAK	AUART	☑	AUART	☐	CHAR	4	Sales Document Type
BSTNK	VBAK	BSTNK	☑	BSTNK	☐	CHAR	20	Customer purchase order number
KUNNR	VBPA	KUNNR	☑	KUNNR	☐	CHAR	10	Customer Number
ERNAM	VBAK	ERNAM	☑	ERNAM	☐	CHAR	12	Name of Person who Created the Object
VBELN	VBAK	VBELN	☑	VBELN_VA	☐	CHAR	10	Sales Document
ADRNR_AG	VBPA	ADRNR	☑	ADRNR	☐	CHAR	10	Address
ADRNR	VBPA	ADRNR	☑	ADRNR	☐	CHAR	10	Address
DATAB	DDDDLCHARTYPES	CDATS	☑		☐	DATS	8	Date Column
DATBI	DDDDLCHARTYPES	CDATS	☑		☐	DATS	8	Date Column
.APPEND	Z_VKP_LI_EXT_CDS		☑		☐		0	
LAND1	VBPA	LAND1	☑	LAND1	☐	CHAR	3	Country Key
			☐		☐			

Attention:
You must also manually activate the top-level view to get the append view attributes into Z_V_VAKPA_CDS.

Dictionary: Display View

DDL SQL View: Z_V_VAKPA_CDS Active
Short Description: compatiblity view for VAKPA
DDL Source: Z_V_VAKPA

Attributes | Table/Join Conditions | View Flds | Selection Conditions | Maint.Status

Table fields

View field	Table	Field	Key	Data elem.	M...	DTyp	Length	Short description
MANDT	Z_V_VAKPA_LI_CDS	MANDT	☑	MANDT	☐	CLNT	3	Client
KUNDE	Z_V_VAKPA_LI_CDS	KUNDE	☑	LIFNR	☐	CHAR	10	Account Number of Vendor or Creditor
PARVW	Z_V_VAKPA_LI_CDS	PARVW	☑	PARVW	☐	CHAR	2	Partner Function
VKORG	Z_V_VAKPA_LI_CDS	VKORG	☑	VKORG	☐	CHAR	4	Sales Organization
TRVOG	Z_V_VAKPA_LI_CDS	TRVOG	☑	TRVOG	☐	CHAR	1	Transaction group
AUDAT	Z_V_VAKPA_LI_CDS	AUDAT	☑	AUDAT	☐	DATS	8	Document Date (Date Received/Sent)
VKBUR	Z_V_VAKPA_LI_CDS	VKBUR	☑	VKBUR	☐	CHAR	4	Sales office
VKGRP	Z_V_VAKPA_LI_CDS	VKGRP	☑	VKGRP	☐	CHAR	3	Sales Group
VTWEG	Z_V_VAKPA_LI_CDS	VTWEG	☑	VTWEG	☐	CHAR	2	Distribution Channel
SPART	Z_V_VAKPA_LI_CDS	SPART	☑	SPART	☐	CHAR	2	Division
AUART	Z_V_VAKPA_LI_CDS	AUART	☑	AUART	☐	CHAR	4	Sales Document Type
BSTNK	Z_V_VAKPA_LI_CDS	BSTNK	☑	BSTNK	☐	CHAR	20	Customer purchase order number
KUNNR	Z_V_VAKPA_LI_CDS	KUNNR	☑	KUNNR	☐	CHAR	10	Customer Number
ERNAM	Z_V_VAKPA_LI_CDS	ERNAM	☑	ERNAM	☐	CHAR	12	Name of Person who Created the Object
VBELN	Z_V_VAKPA_LI_CDS	VBELN	☑	VBELN_VA	☐	CHAR	10	Sales Document
ADRNR_AG	Z_V_VAKPA_LI_CDS	ADRNR_AG	☑	ADRNR	☐	CHAR	10	Address
ADRNR	Z_V_VAKPA_LI_CDS	ADRNR	☑	ADRNR	☐	CHAR	10	Address
DATAB	Z_V_VAKPA_LI_CDS	DATAB	☑		☐	DATS	8	
DATBI	Z_V_VAKPA_LI_CDS	DATBI	☑		☐	DATS	8	
LAND1	Z_V_VAKPA_LI_CDS	LAND1	☑	LAND1	☐	CHAR	3	Country Key
			☐		☐			
			☐		☐			
			☐		☐			

For the top-level view, the extending attribute LAND1 is explicitly added to the view.

In the runtime object of the view the attribute LAND1 is also added to the end.

Display: Active runtime object for Z_V_VAKPA_CDS

Database View	Runtime Object Time Stamp
Z_V_VAKPA_CDS	10.08.2015 16:59:54
ABAP Time Stamp	**DYNP Time Stamp**
10.08.2015 16:59:54	10.08.2015 16:59:54

Header of active runtime object

Obj...	Dat...	No...	Tabl...	No...	Ke...	Po...	Ali...	Buf...	Nu...	Le...	Flag 1	P...	DB	Flag 2	Flag 3	Flag 4	Flag 5	Flag 6	Poi...	Uni...	UUID	Leaf...	Con
J	T	20	268	20	268	1	2		0	0	00010110		B	00000000	00000000	00000000	00000000	00000000	0	2	A3DBC255D3D9B...	0	000

Fields of active runtime object

Field Name	Fiel...	De...	Data...	ABA...	DB len...	De...	Field...	Ou...	Fiel...	AB...	AB...	Dic...	Flag 1	Flag 2	Flag 3	Flag 4	Data eleme...	Pre...	R...	R...	Conv...	SET/...	Check t...
MANDT	1	0	CLNT	6	6	0	0	3	152	0	C	0	00000001	00001000	00000010	00000000	MANDT	0					T000
KUNDE	2	0	CHAR	20	20	0	6	10	40	0	C	0	00100001	00001110	00000010	00000000	KUNNR	0			ALP...	KUN	KNA1
PARVW	3	0	CHAR	4	4	0	26	2	40	0	C	0	00100001	00001110	00000010	00000000	PARVW	0			PAR...	PAR	TPAR
VKORG	4	0	CHAR	8	8	0	30	4	40	0	C	0	00100001	00001100	00000010	00000000	VKORG	0				VKO	TVKO
TRVOG	5	0	CHAR	2	2	0	38	1	40	0	C	0	00010001	00000100	00000010	00000000	TRVOG	0				VTV	
AUDAT	6	0	DATS	16	16	0	40	10	156	1	D	0	00100001	00000000	00000010	00000000	AUDAT	0					
VKBUR	7	0	CHAR	8	8	0	56	4	40	0	C	0	00100001	00001100	00000010	00000000	VKBUR	0				VKB	TVKBZ
VKGRP	8	0	CHAR	6	6	0	64	3	40	0	C	0	00100001	00001100	00000010	00000000	VKGRP	0				VKG	TVBVK
VTWEG	9	0	CHAR	4	4	0	70	2	40	0	C	0	00100001	00001100	00000010	00000000	VTWEG	0				VTW	TVKOV
SPART	10	0	CHAR	4	4	0	74	2	40	0	C	0	00100001	00001100	00000010	00000000	SPART	0				SPA	TVTA
AUART	11	0	CHAR	8	8	0	78	4	40	0	C	0	00000001	00001110	00000010	00000000	AUART	0			AUA...	AAT	TVAK
BSTNK	12	0	CHAR	40	40	0	86	20	40	0	C	0	00000101	00000000	00000000	00000000	BSTNK	0					
KUNNR	13	0	CHAR	20	20	0	126	10	40	0	C	0	00100001	00001110	00000010	00000000	KUNNR	0			ALP...	KUN	KNA1
ERNAM	14	0	CHAR	24	24	0	146	12	40	0	C	0	00000001	00000000	00000000	00000000	ERNAM	0					
VBELN	15	0	CHAR	20	20	0	170	10	40	0	C	0	00000001	00000110	00000011	00000000	VBELN_VA	0			ALP...	AUN	
ADRNR_AG	16	0	CHAR	20	20	0	190	10	40	0	C	0	00000001	00000010	00000000	00000000	ADRNR	0			ALP...		
ADRNR	17	0	CHAR	20	20	0	210	10	40	0	C	0	00000001	00000010	00000000	00000000	ADRNR	0			ALP...		
DATAB	18	0	DATS	16	16	0	230	10	156	1	D	0	00000001	00000000	00000010	00000000		0					
DATBI	19	0	DATS	16	16	0	246	10	156	1	D	0	00000001	00000000	00000010	00000000		0					
LAND1	20	0	CHAR	6	6	0	262	3	40	0	C	0	00100001	00001100	00000010	00000000	LAND1	0				LND	T005

Example 2:

In addition to Example 1, the attribute HITYP (customer hierarchy type) should only be added for the role ‘customer’ , while for the partner role ‘supplier’, it should remain as the initial one.

In order to achieve this, both lower-level views must be extended by the additional attribute HITYP.

Extension view for customers:

```
@AbapCatalog.sqlViewAppendName: 'Z_VKP_KU_EXT_CDS'
@EndUserText.label: 'extend view for sub-view customer'
extend view Z_V_VAKPA_KU with Z_V_VAKPA_KU_EXT {
    vbpa.land1 as land1,
    vbpa.hityp as hityp

}
```

For the customer view, the additional attribute can be added to LAND1 in a similar fashion:

Extension view for suppliers:

```
@AbapCatalog.sqlViewAppendName: 'Z_VKP_LI_EXT_CDS'
@EndUserText.label: 'extend view for sub-view customer'
extend view Z_V_VAKPA_LI with Z_V_VAKPA_LI_EXT {
    vbpa.land1 as land1,
    cast ('' as HITYP_KH) as hityp

}
```

Since for suppliers the attribute HITYP should remain as initial, the initial value ‘’ must be cast to refer to data element HITYP_KH as the target attribute HITYP ,to make sure that both attribute lists return an identical structure.

In the runtime object for the top-level view, the attribute HITYP will be added to the end:

Display: Active runtime object for Z_V_VAKPA_CDS

Database View	Runtime Object Time Stamp
Z_V_VAKPA_CDS	10.08.2015 17:18:42
ABAP Time Stamp	**DYNP Time Stamp**
10.08.2015 17:18:42	10.08.2015 17:18:42

Header of active runtime object

Obj...	Dat...	No...	Tabl...	No...	Ke...	Po...	Ali...	Buf...	Nu...	Le...	Flag 1	P...	DB	Flag 2	Flag 3	Flag 4	Flag 5	Flag 6	Poi...	Uni...	UUID	Leaf...	Com
J	T	21	270	21	270	1	2		0	0	00010110		B	00000000	00000000	00000000	00000000	00000000	0	2	71D7C3550E3078...	0	000

Fields of active runtime object

Field Name	Fiel...	De...	Data...	ABA...	DB len...	De...	Field...	Ou...	Fiel...	AB...	AB...	Dic...	Flag 1	Flag 2	Flag 3	Flag 4	Data eleme...	Pre...	R...	R...	Conv...	SET/...	Check t...
MANDT	1	0	CLNT	6	6	0	0	3	152	0	C	0	00000001	00001000	00000010	00000000	MANDT	0					T000
KUNDE	2	0	CHAR	20	20	0	6	10	40	0	C	0	00100001	00001110	00000010	00000000	KUNNR	0			ALP...	KUN	KNA1
PARVW	3	0	CHAR	4	4	0	26	2	40	0	C	0	00100001	00001110	00000010	00000000	PARVW	0			PAR...	PAR	TPAR
VKORG	4	0	CHAR	8	8	0	30	4	40	0	C	0	00100001	00001100	00000010	00000000	VKORG	0				VKO	TVKO
TRVOG	5	0	CHAR	2	2	0	38	1	40	0	C	0	00010001	00000100	00000010	00000000	TRVOG	0				VTV	
AUDAT	6	0	DATS	16	16	0	40	10	156	1	D	0	00100001	00000000	00000010	00000000	AUDAT	0					
VKBUR	7	0	CHAR	8	8	0	56	4	40	0	C	0	00100001	00001100	00000010	00000000	VKBUR	0				VKB	TVKBZ
VKGRP	8	0	CHAR	6	6	0	64	3	40	0	C	0	00100001	00001100	00000010	00000000	VKGRP	0				VKG	TVBVK
VTWEG	9	0	CHAR	4	4	0	70	2	40	0	C	0	00100001	00001100	00000010	00000000	VTWEG	0				VTW	TVKOV
SPART	10	0	CHAR	4	4	0	74	2	40	0	C	0	00100001	00001100	00000010	00000000	SPART	0				SPA	TVTA
AUART	11	0	CHAR	8	8	0	78	4	40	0	C	0	00000001	00001110	00000010	00000000	AUART	0			AUA...	AAT	TVAK
BSTNK	12	0	CHAR	40	40	0	86	20	40	0	C	0	00000101	00000000	00000000	00000000	BSTNK	0					
KUNNR	13	0	CHAR	20	20	0	126	10	40	0	C	0	00100001	00001110	00000010	00000000	KUNNR	0			ALP...	KUN	KNA1
ERNAM	14	0	CHAR	24	24	0	146	12	40	0	C	0	00000001	00000000	00000000	00000000	ERNAM	0					
VBELN	15	0	CHAR	20	20	0	170	10	40	0	C	0	00000001	00000110	00000011	00000000	VBELN_VA	0			ALP...	AUN	
ADRNR_AG	16	0	CHAR	20	20	0	190	10	40	0	C	0	00000001	00000010	00000000	00000000	ADRNR	0			ALP...		
ADRNR	17	0	CHAR	20	20	0	210	10	40	0	C	0	00000001	00000010	00000000	00000000	ADRNR	0			ALP...		
DATAB	18	0	DATS	16	16	0	230	10	156	1	D	0	00000001	00000000	00000010	00000000		0					
DATBI	19	0	DATS	16	16	0	246	10	156	1	D	0	00000001	00000000	00000010	00000000		0					
LAND1	20	0	CHAR	6	6	0	262	3	40	0	C	0	00100001	00001100	00000010	00000000	LAND1	0				LND	T005
HITYP	21	0	CHAR	2	2	0	268	1	40	0	C	0	00000001	00001000	00000010	00000000	HITYP_KH	0					THIT