**Selective data deletion in adso**

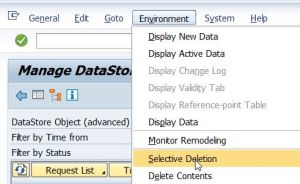
**+ Sap bw selective data deletion in adso**

<https://www.dahlbeer.com/post/selective-deletion-from-advanced-dso-advanced-data-store-object>

The Advanced DSO consolidates the functionality of a DataStore Object, a PSA, an InfoCube and a Hybridprovider into a single object and completely replaces these objects in data models built in BW 7.4 SP8 and onwards.

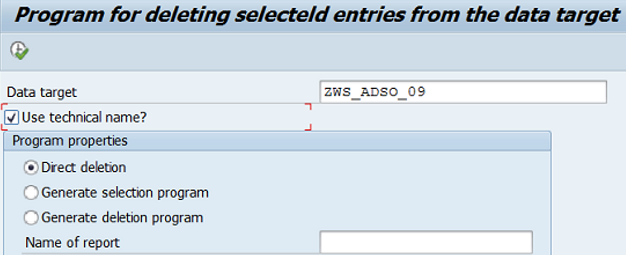
ADSO = DSO + PSA + InfoCube + Hybridprovider.

In the menu of ADSO there is actually a button *Selective deletion*. You can find it in the main menu when you already pressed *manage* for the ADSO



This option is new and gives often errors, so there is an alternative way to perform it.

Selective deletion from ADSO can be also done with a standard ABAP program. In transaction ***SE38***, you need to execute program ***RSDRD\_DELETE\_FACTS***, select the ADSO, enter the selection criteria and execute.



*Rem*

If You choose *Generate deletion program*, you can set a name and use the program later. The deletion program can be used in *process chains* too.

Another transaction ***DELETE\_FACTS***, uses the same ABAP program for selective deletion form an ADSO.

### **Deleting overlapping requests**

<https://blog.maruskin.eu/2020/02/deleting-overlapping-requests.html>

Depending on BW version a Notes like [1336410](https://launchpad.support.sap.com/#/notes/1336410) - 70SP22 Enhancements to CL\_RSBK\_DTP=>GET\_ALL\_BY\_PROPERTY and [1359397](https://launchpad.support.sap.com/#/notes/1359397) - P22:PC:REQUDEL Switching full variant to deleting delta DTPs need to be implemented. Once it is the case, the *Delete Overlapping Requests* process works with delta enabled DTPs. Only drawback is that the DTPS must be set with indicator *specifying that delta data is to be transferred only once*.

**Обзор по теме**

<https://www.tcodesearch.com/sap-fms/detail?id=RSDRD_SEL_DELETION>

# RSDRD\_SEL\_DELETION SAP Function module - Selective deletion in data targets

<https://www.se80.co.uk/sapfms/r/rsdr/rsdrd_sel_deletion.htm>

*Propagate Deletion* – Boolean type used in SAP BW. The associated constants are *RS\_C\_TRUE* and *RS\_C\_FALSE*.

*RS\_C\_TRUE Propagate Deletion*

i\_del\_activ = RS\_C\_TRUE - Deletion from Active Version - /BIC/ABDNPSLPM2

i\_del\_update = RS\_C\_FALSE - Deletion from Update Table - /BIC/ABDNPSLPM1

i\_del\_change = RS\_C\_FALSE - Deletion from Change-Log - /BIC/ABDNPSLPM3

RS\_C\_TRUE *Propagate Deletion* what it is mean

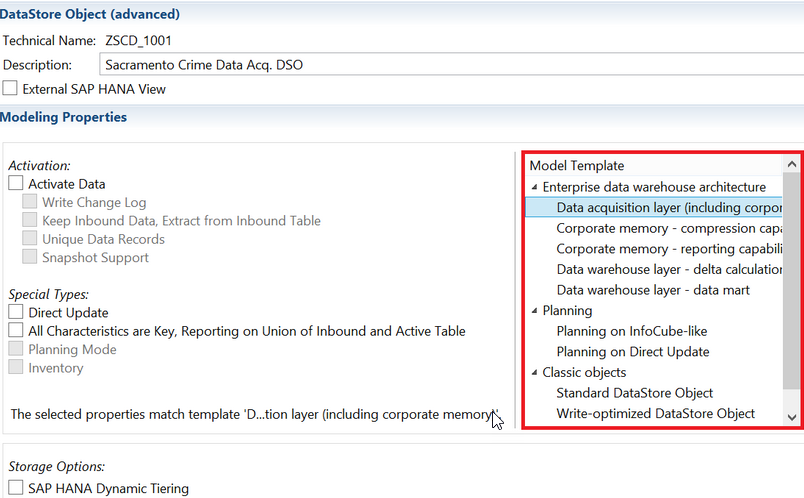
# All you need to know about Advanced DSOs

<https://www.just-bi.nl/all-you-need-to-know-about-adso/>

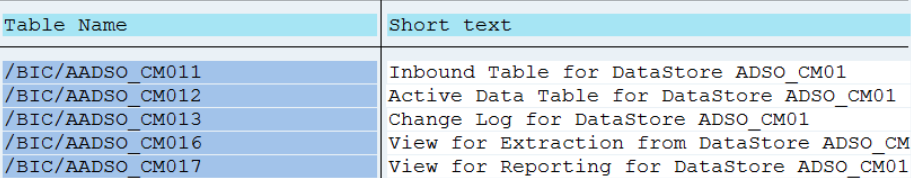
ADSO Model Template

There are 3 Model Template categories

* Enterprise data warehouse architecture
* Planning
* Classic Objects



All ADSOs have the *system tables* listed below, but depending on the ADSO, they might remain unused.



## Категория - *Enterprise data warehouse architecture*

### ***Шаблон*** - ***Data Acquisition layer*** /including corporate memory/.

*Data Acquisition /*ækwɪˈzɪʃn *сбор, получение/ Layer*

*Inbound Table PSA*

ADSOs modeled using the *Data Acquisition Layer* template *don’t use an Active Table*, just like a Write-Optimized DSO. All the records in the *Inbound Table* contain a Request Transaction Number - *TSN*, Data packet, and Data record number.

You can use this model template as a persistent staging area - *PSA* and in the 1st layer of the EDW architecture.

*Rem*

The *Inbound Table* - is just another SAP term for the *New Data* / *Activation Queue Table*.

### ***Шаблон*** - ***Corporate Memory – compression capabilities***

*Corporate Memory – compression capabilities*

*Inbound Table /the data is erased/ Active Table*

*Extracting*

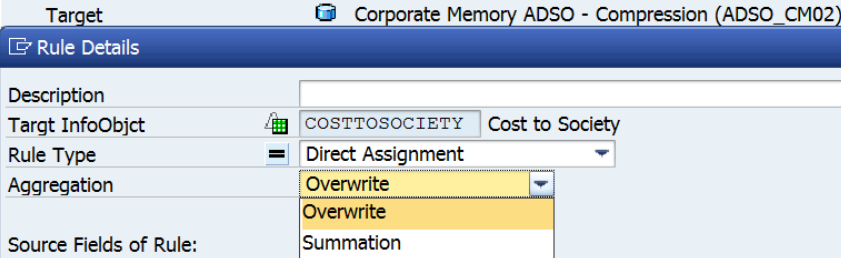
*Reporting*

The *compression ADSO* *doesn’t use a Change Log table* - only an Inbound and an Active data table. As soon as a load request is activated, the system loads the data into the *Active* table and *deletes it from the Inbound Table*.

Reporting in the CM – Compression ADSO takes place on the Active table. ***Extraction*** - *on both* - its *Active* and *Inbound* tables.

If there are 2 records with the same key, BW/4HANA

* overwrites all the *characteristics* of the record with the characteristics of the lastly loaded record;
* either overwrites or sums the *Key Figures*, depending on what you select in the transformation rules

.

Use this template if for instance - *the data is so old* that you don’t need to trace it back.

### ***Шаблон*** - ***Corporate Memory – reporting capabilities***

*Corporate Memory – reporting capabilities*

*Inbound Table /the data isn’t erased/ Active Table*

*Extracting*

*Reporting*

The option *Keep Inbound Data, Extract from Inbound Table* is turned on ⇒ when these kinds of ADSOs are activated, the sytem *does not erase data from the Inbound table*.

The data is *extracted*from *the Inbound table/*

These ADSOs are a good solution when you not using a Change Log, but need *to be able to track back records to their specific load*.

### ***Шаблон*** - ***Data warehouse layer – delta calculation***

*Data warehouse layer – delta calculation*

*Inbound Table /the data is erased/ Active Table Change Log Table*

*full loads delta extractions*

*Reporting*

These types of ADSOs have a *Change Log table for delta extractions*, an Inbound and an Active data tables - the latter is used for reporting and *full loads*.

* The *Inbound* table is empty after activation.
* In the *Change Log* table we can find the new, before- and after status of records in column R.

### ***Шаблон*** - ***Data warehouse layer – data mart*** /mɑːt витрина/

The ADSO behave just like an InfoCube - it does not have a Change Log. The *Inbound* table functions as a Cube’s *F-table* and the *Active Data* table - as the *E-table* for compressed data /similar to the InfoCube function Collapse/.

After activation, the Inbound table is empty

*Data warehouse layer – data mart*

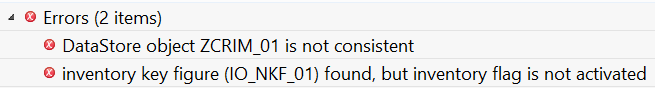
*Inbound Table /the data is erased/ Active Table*

*F-table*  *E-table*

The system unions both the Inbound and Active tables together for reporting and extraction.

Thanks to HANA’s in-memory capabilities however, we no longer need dimension tables - BW now writes *Master data identifiers* - ***SID***s directly to the Fact table.

This makes InfoCubes more or less obsolete. You still need to model ADSOs like an InfoCube if you use *non-cumulative Key Figures*, but to achieve that you would also need to manually tick the *Inventory* property box. If you do not tick this box, you’ll get an error if you try to add a non-cumulative Key Figure to your ADSO -

.

The option *All characteristics are Key, Reporting on Union of Inbound and Active Table* - is a prerequisite to use non-cumulative Key Figures.

# − Is DSO Activation without writing a Changelog possible?

− You can activate an ADSO without needing a change log - this will just create the *SIDs* required for reporting using old data models like Cube or OLAP based reporting.

# Snapshot generation in HR reporting with ADSOs

<https://www.btelligent.com/en/blog/snapshot-generation-in-hr-reporting-with-adsos/>

Use of HANA for SAP BW (from 7.5 SP4) results in a new possibility - The ability to utilize an ADSO with a ***snapshot*** function. Similar to SLT/RS or a HANA-based change data capture, this ADSO setting fills the change log table based on a full load, and makes it possible to also *consider deletions during further delta processing*. In this way, changes to data can be clearly detected and saved on a persistent basis in the corporate memory layer for later evaluation. In addition, filling of time-dependent attributes of InfoObjects can be accelerated, because this process continues to running wholly or partly in the ABAP stack, and is therefore not the fastest loading type.

## [Activating the snapshot function](https://www.btelligent.com/en/blog/snapshot-generation-in-hr-reporting-with-adsos/)

The following screenshot shows how snapshot support can be activated in the modeling attributes /it should be noted that ADSO must then always be loaded completely/.



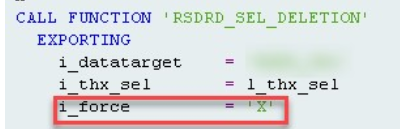
## [Delta behavior - RECORDMODRE in the change log](https://www.btelligent.com/en/blog/snapshot-generation-in-hr-reporting-with-adsos/)

The appropriately set ADSO with snapshot support generates entries in the change log. However, a problem is that after a data record has been deleted, no entry is created with RECORDMODE 'D' for deletion in the change log, but instead a reverse image with RECORDMODE 'R'. This contains all the information of the original data record, and would thus not reset the attributes of an InfoObject. For this reason, the data record with the reverse image must be modified to a deletion when filling an InfoObject. For this, all non-key columns - with the exception of the start date of validity - must be deleted. This is needed in order to overwrite the correct time range from the InfoObject. This is because if transaction data are present, master data can no longer be deleted. The records need to be "emptied".

# ADSO Deletion Program(Inbound Table) in SAP BW

<https://blogs.sap.com/2022/01/20/adso-deletion-programinbound-table-in-sap-bw/>

When we use ADSO and wanted to delete *Inbound table*  we need to pass one more variant in export as given below



propagate deletion at adso selective data deletion

BI\_ADSO propagate deletion option in adso selective deletion

*Z*GP00O2TIHH04Z7UZ70NZ4S27ZQT

SELECT COUNT( \* ) AS "CNT"

FROM "/BIC/ABDNPSLPM2" "A"

WHERE ("A"."DATE0" BETWEEN '20180101' AND '20181231')