

# Analysis Process Designer: Step by Step Process for Formatting the Query Extract



## Applies to:

SAP BW 3.x & SAP BI Net Weaver 2004s. For more information, visit the [EDW homepage](#).

## Summary

This document gives step by step procedure to format the output file using APD.

**Author:** Mayuri Sinha

**Company:** Infosys Technologies Limited

**Created on:** 24 August 2010

## Author Bio

Mayuri Sinha is working as a Technology Analyst with Infosys Technologies Limited.

## Table of Contents

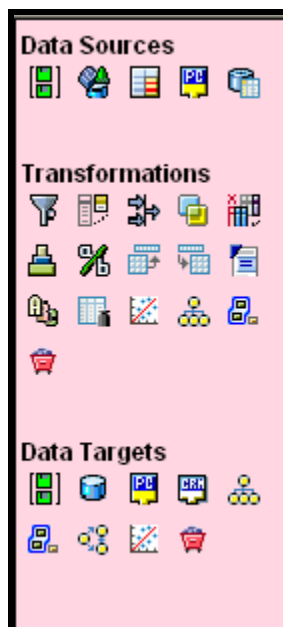
Introduction .....	3
Live Scenario .....	4
Step 1: .....	5
Step 2: .....	6
Step 3: .....	6
Step 4: .....	7
Step 5: .....	7
Step 6: .....	9
Step 7: .....	9
Step 8: .....	10
Step 9: .....	10
Related Content .....	11
Disclaimer and Liability Notice .....	12

## Introduction

The Analysis Process Designer is the application environment for the SAP data mining solution. The Analysis Process Designer (APD) makes it possible to find and identify the hidden or complex relationships between data in a simple way. The APD workbench provides an intuitive graphical interface that enables you to visualize, transform, and deploy data from your business warehouse. Various data transformations are provided for this purpose, such as statistical and mathematical calculations, and data cleansing or structuring processes.

An APD generally consists of three parts:

- **Data Sources:** Read data from InfoProviders, Attributes of a Characteristic, Query, Flat Files, Database Tables
- **Transformations:** Filter restricted amount of data, Aggregate Data, Join data from different Data Sources, ABAP Routines, Formula, Sort data, etc.
- **Data Targets:** Write data directly to Data Store Objects (DSOs), Flat files, Update CRM Attributes, Change attributes of a Characteristics, etc.



## Live Scenario

Here we need to provide business with extract of a data of a query in text format in well formatted condition.

The formatting will include the below:

- Changing the Date Format.
- Article length should be 8
- Volume and Sales Amount should not have spaces at the end.

Output if we execute the APD without formatting

```
Item,Date,Sales,Volume
000000000010000020,20081110,44.20 ,8.000 ,
000000000010000020,20081111,34.00 ,6.000 ,
000000000010000020,20081112,106.86 ,20.000 ,
000000000010000020,20081113,28.56 ,6.000 ,
000000000010000020,20081114,13.60 ,3.000 ,
000000000010000020,20081115,6.80 ,2.000 ,
000000000010000020,20081117,40.80 ,11.000 ,
000000000010000020,20081118,27.20 ,4.000 ,
000000000010000020,20081119,23.80 ,6.000 ,
```

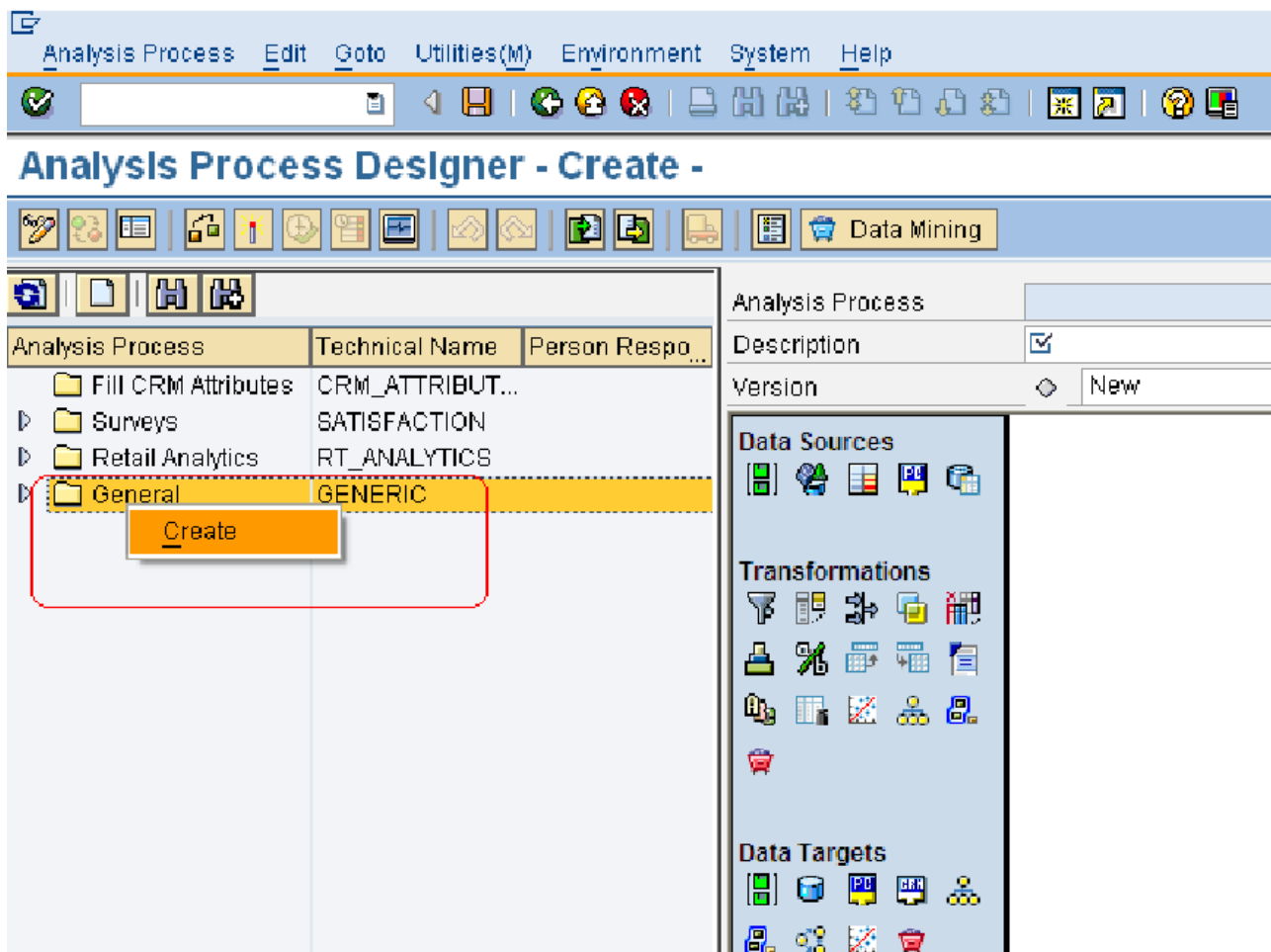
Output if we execute the APD after formatting

```
Item,Date,Volume,Sales
10000020,10.11.2008,8,51.93,
10000020,11.11.2008,6,39.95,
10000020,12.11.2008,20,125.56,
10000020,13.11.2008,6,33.56,
10000020,14.11.2008,3,15.98,
10000020,15.11.2008,2,7.99-,
10000020,17.11.2008,11,47.94,
10000020,18.11.2008,4,31.96,
10000020,19.11.2008,6,27.96,
10000020,20.11.2008,2,15.98,
10000020,21.11.2008,4,7.99,
10000020,24.11.2008,1,7.99,
10000020,25.11.2008,27,31.96,|
```

In the below sections we will have a look as to how can we achieve this.

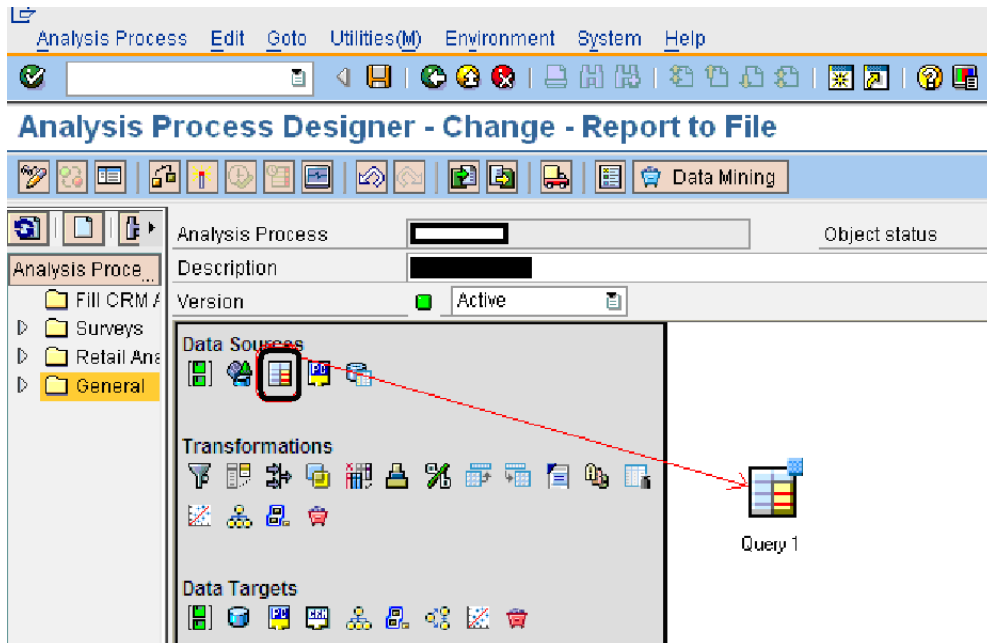
**Step 1:**

Go to TCode RSANWB and create a new APD in the General Node



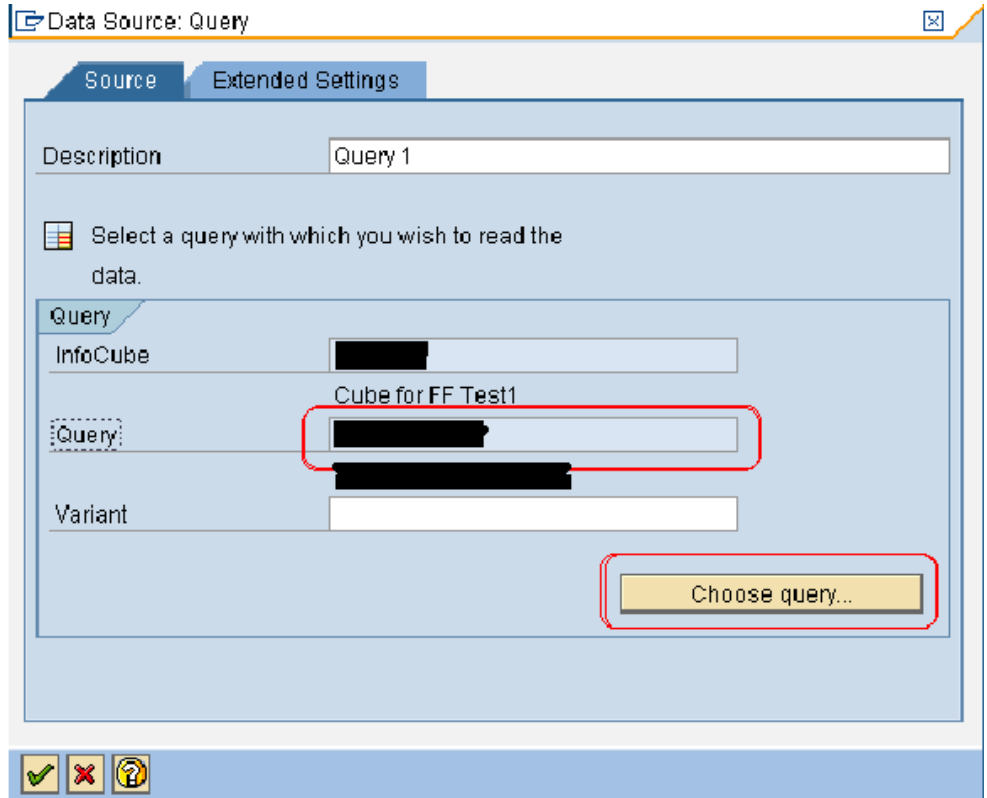
**Step 2:**

As the source of the data is a query, from the data source section we select a query. This will enable the source of data to be extracted from the query.



**Step 3:**

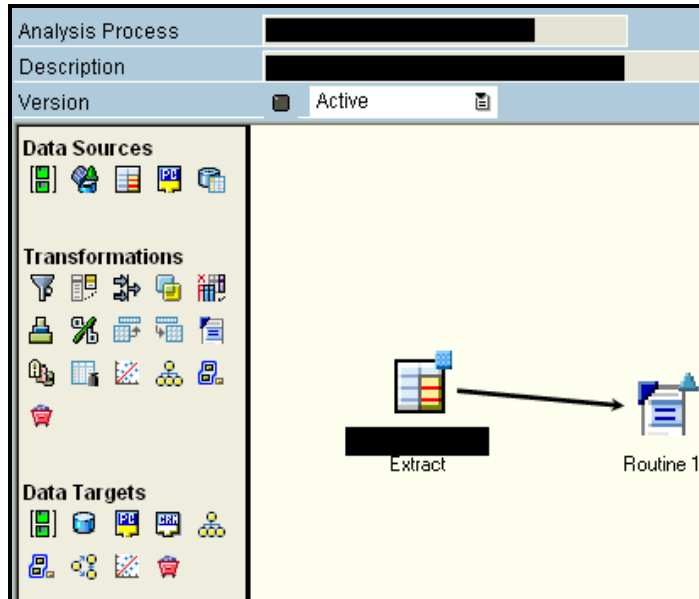
Specify the properties. In this section we will specify the query name from which we want to extract the data.



**Step 4:**

As we want to format the data we will choose the transformation of type routine.

Here we will write an ABAP program to format the output. Here we also link the query to the routine.

**Step 5:**

We write the ABAP code as below to format the output.

```
REPORT RSAN_WB_ROUTINE_TEMP_REPORT .
TYPES: BEGIN OF Y_SOURCE_FIELDS ,
        ARTICLE TYPE /BI0/OIARTICLE ,
        CALDAY TYPE /BI0/OICALDAY ,
        KYF_0004 TYPE FLOAT ,
        KYF_0001 TYPE I ,
        KYF_0003 TYPE FLOAT ,
        KYF_0002 TYPE FLOAT,
        END OF Y_SOURCE_FIELDS .
TYPES: YT_SOURCE_FIELDS TYPE STANDARD TABLE OF Y_SOURCE_FIELDS .
TYPES: BEGIN OF Y_TARGET_FIELDS ,
        ARTICLE(8) TYPE C ,
        CALDAY(10) TYPE C ,
        VOLUME(18) TYPE C ,
        SALES(18) TYPE C ,
        END OF Y_TARGET_FIELDS .
TYPES: YT_TARGET_FIELDS TYPE STANDARD TABLE OF Y_TARGET_FIELDS .
*----- BEGIN OF TYPE DEFINITIONS -----
*TYPES: ...
*----- END OF TYPE DEFINITIONS -----
FORM COMPUTE_DATA_TRANSFORMATION
    USING      IT_SOURCE TYPE YT_SOURCE_FIELDS
              IR_CONTEXT TYPE REF TO IF_RSAN_RT_ROUTINE_CONTEXT
    EXPORTING ET_TARGET TYPE YT_TARGET_FIELDS .
*----- BEGIN OF TRANSFORMATION CODE -----
DATA: LS_SOURCE TYPE Y_SOURCE_FIELDS,
      LS_TARGET TYPE Y_TARGET_FIELDS,
```

```

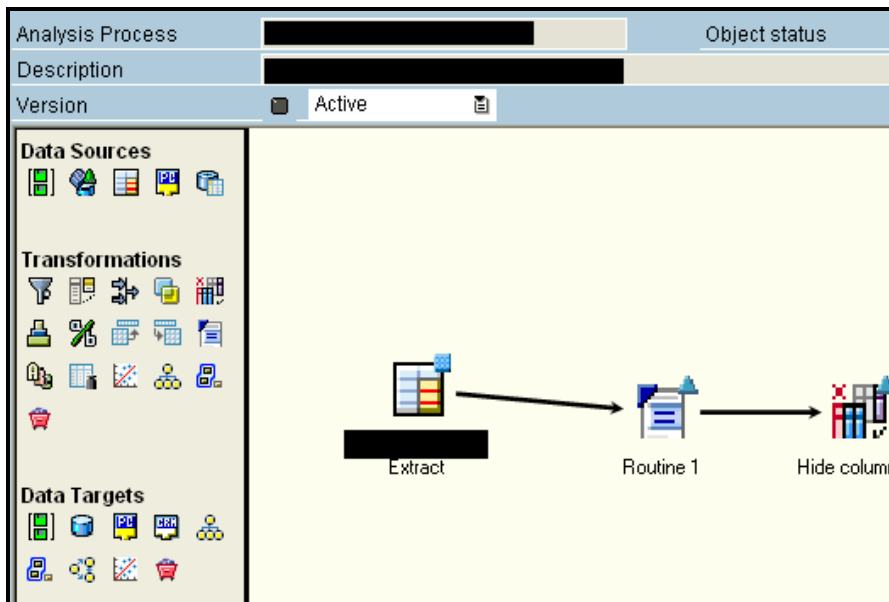
    LV_CALDAY(10) TYPE C,
    LV_VOLUME TYPE I,
    L_LEN TYPE I,
    L_LENGTH TYPE N.
LOOP AT IT_SOURCE INTO LS_SOURCE.
  MOVE-CORRESPONDING LS_SOURCE TO LS_TARGET.
  CONCATENATE LS_SOURCE-CALDAY+6(2) '.' LS_SOURCE-CALDAY+4(2) '.'
  LS_SOURCE-CALDAY(4) INTO LV_CALDAY.
  LS_TARGET-CALDAY = LV_CALDAY.
*ARTICLE SHOULD BE LEFT PADDED WITH ZEROES SO THAT IT IS 8 CHARACTERS
*LONG.
  IF STRLEN( LS_SOURCE-ARTICLE) GT 8.
    SHIFT LS_SOURCE-ARTICLE BY 8 PLACES LEFT.
  ENDIF.
  L_LEN = STRLEN( LS_SOURCE-ARTICLE ).
  IF L_LEN GT 8.
    L_LEN = L_LEN - 8.
  ELSE.
    L_LEN = 0.
  ENDIF.
  L_LENGTH = L_LEN.
  LS_TARGET-ARTICLE = LS_SOURCE-ARTICLE+L_LENGTH(8) .
  LV_VOLUME = LS_SOURCE-KYF_0001.
  LS_TARGET-VOLUME = VOLUME.
  SHIFT LS_TARGET-VOLUME RIGHT DELETING TRAILING SPACE.
  CONDENSE LS_TARGET-VOLUME NO-GAPS.
  LS_TARGET-SALES = LS_SOURCE-KYF_0002.
  SHIFT LS_TARGET-SALES RIGHT DELETING TRAILING SPACE.
  CONDENSE LS_TARGET-SALES NO-GAPS.
  APPEND LS_TARGET TO ET_TARGET.
ENDLOOP.
*----- END OF TRANSFORMATION CODE -----
ENDFORM.

```

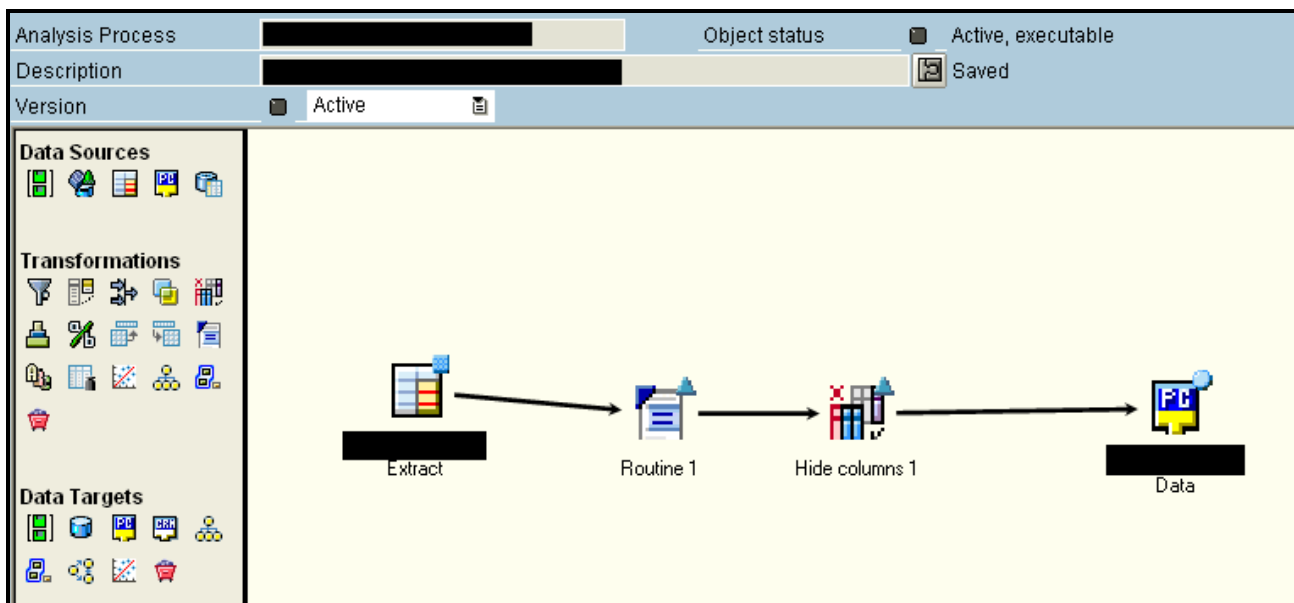


**Step 6:**

We then drag in the transformation 'Hide/Show Columns' . With the help of this we can specify the sequence in which we want to have the output.

**Step 7:**

After having done all the formatting changes , we need to specify the output format and location. This can be done by dragging the data target Flat File.



**Step 8:**

We can specify the properties of the output file as shown below.

**Data Target: File**

**Data Target** | **CSV File Properties**

Description: Data Target: File 1

Specify Where the File Should Be Generated

Write Data to

Client Workstation

Application Server

Logical File Name

File: C:\Documents and Settings\Desktop\KPI\_FILE.csv

Extended Settings

Write Mode: Create File Again. If Already Available. Overwrite

Insert Header Row with Field Names in the First Row of the File

✓ ✗ ?

**Data Target: File**

**Data Target** | **CSV File Properties**

Separators: ,

Separator Field Name/InfoObject: ⋮

**Step 9:**

Now as a final step we need to save and activate the APD. When we execute this APD this will give us a formatted output.

## Related Content

[Analysis Process Designer](#)

[Creating, Changing, and Activating a Model](#)

[Checking Data](#)

## 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.