

Technical Documentation - CERISE

- 1 [Access Management](#)
- 2 [DataFlow](#)
 - 2.1 [Overview](#)
 - 2.2 [Technical Rules on Workbench](#)
 - 2.2.1.1 [Further explanations for program ZBW_ZECO2_FILE - ZECO2_FILE Transaction](#)
 - 2.2.1.1.1 [Source Code](#)
 - 2.2.1.1.1.1 [Interface](#)
 - 2.2.1.1.1.2 [Import File](#)
 - 2.2.1.1.1.3 [Program Output](#)
 - 2.2.1.1.1.4 [Program Load](#)
 - 2.2.1.1.1.5 [Program Log](#)
 - 2.2.2 [Historisation of Conversion Data](#)
 - 2.3 [Reporting](#)
 - 2.4 [Dependencies with other applications](#)
- 3 [Data Loading](#)
 - 3.1 [Info Providers and objects loaded](#)
 - 3.2 [Loading frequency](#)
 - 3.3 [Average performance](#)
 - 3.4 [Record Keeping](#)
 - 3.4.1 [Global filters \(C_GLBFLT\) used in Chains](#)
 - 3.4.2 [ABAP Programs used for Data Load](#)
 - 3.4.2.1 [Further explanations for program ZECO2_LOAD_VIEWS - ZECO2 Transaction - ZECR Function Group](#)
- 4 [Data Quality Control](#)
- 5 [Operational Documentation](#)
 - 5.1 [Procedures](#)
 - 5.2 [Scheduling](#)
 - 5.3 [Monitoring](#)
 - 5.4 [Error Handling](#)
 - 5.5 [Known Bugs](#)
 - 5.6 [Roadmap](#)

Access Management

Roles & Access

List of application role + menu role and explanation if we have several applications role with specials rules.

Role Code	Role Description	Explanation
ZBI_RCS_E CO_A01	CO2 - Primary Energy & CO2 Reporting - End user	End user role
ZP2_RCS_E CO_A01	CO2 - Primary Energy & CO2 Reporting - End user	End user role
ZR_RCS_CA _M72	CO2 - Primary Energy & CO2 Reporting	Gives access to queries & workbooks CO2
ZR_RCS_C A_M721	CO2 - Transaction ZECO2	Gives access to transaction ZECO2 which allows manual entries through SM30 (Screens in Function Group ZECR) and flat file import through transaction ZECO2_FILE.

Authorization Objects

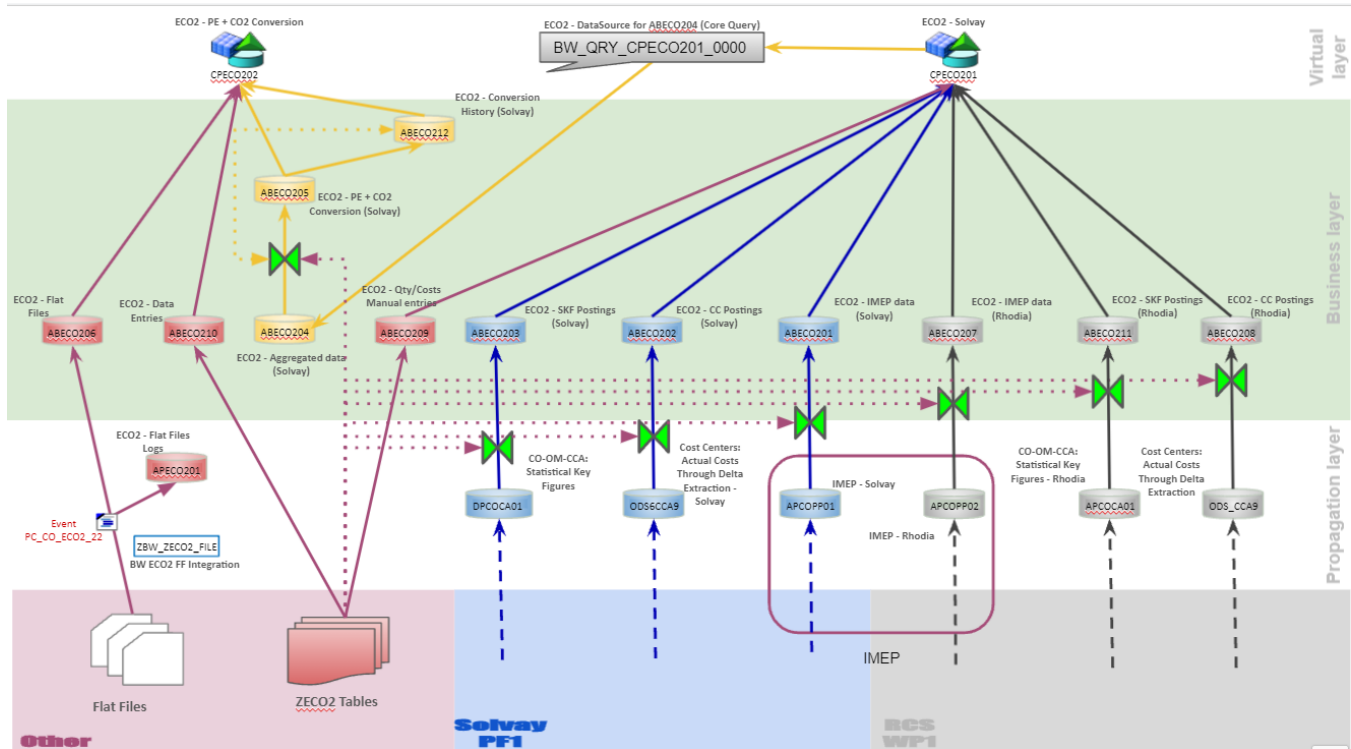
List of authorization objects mandatory for the application.

Authorization object	Explanation
C_AUTHMA	Authorization scope: ZR_*_CA_P00
CPFCTR1_2	GBU: ZR_*_CA_P05

https://docs.google.com/spreadsheets/d/10GEfKYqrT1eeTO_uHYAheL1GX7L5y_pvH0KQU64qh5I/edit#gid=131158862

DataFlow

Overview



<https://drive.google.com/drive/folders/1VxDY1lNYfCd9nwPFQZOclvSZitsbUro>

Technical Rules on Workbench

TRSF: ODSO DPCOCA01 -> ADSO ABECO203 & TRSF: ADSO APCOCA01 -> ADSO ABECO211 & TRSF: APCOPP01 -> ADSO ABECO201 & TRSF: APCOPP02 -> ABECO207 & TRSF: ODSO ODS6CCA9 -> ADSO ABECO202 & TRSF: ODS_CCA9 -> ABECO208

Start routine:

Collecting Information from Table T006: The code performs a SELECT query to extract information from the T006 table into the internal table ITB_T006. The extracted data includes rows where the DIMID field is equal to specific values like 'AAAADL', 'ENERGY', 'MASS', 'POWER', 'VOLUME', or 'ZAI'.

Checks if the unit extracted from the source structure is found in the ITB_T006 table. If the unit is not found in the table, the data row is deleted from the source package.

Collecting Cost Center Information: Next, the code performs two SELECT queries to collect information about cost centers from the ZECR_FT_CC_T and / BIC/MC_COSTCTR tables. It uses this information to potentially remove non-compliant data rows.

If source line exists in ITB_CC (in function of co_area and c_cosctr):

Check in ITB_FT_CC is the logsys and c_costctr exist. If not, the record is deleted.

If source line not exists in ITB_CC (in function of co_area and c_cosctr):

Delete record.

Collecting Other Information: The code carries out additional SELECT queries to collect information from various tables such as /BIC/PC_COSTCTR, /BIC/PC_COMPCDE, ZECR_AS_CC2EQ_T, ZECR_AS_CE2VC_T, ZECR_CV_UOM2_T, ZECR_AS_VFLAG_T, and /BIC/PC_PRDEQUI.

Collecting Information from Table T006: The code performs a SELECT query to extract information from the T006 table into the internal table ITB_T006. The extracted data includes rows where the DIMID field is equal to specific values like 'AAAADL', 'ENERGY', 'MASS', 'POWER', 'VOLUME', or 'ZAI'.

Checks if the unit extracted from the source structure is found in the ITB_T006 table. If the unit is not found in the table, the data row is deleted from the source package.

End routine:

Read internal table ITB_UOM (from table ZECR_CV_UOM2_T) with help of LOGSYS, COSTELMNT, U_UNITUM, C_PLANT, and C_COSTCTR. If a correspondance is done and in function of the period: K_ACT_QTY * <FS_UOM>-K_UOMCVRT * <FS_UOM>-K_UMREZ / <FS_UOM>-K_UMREN and U_UNITUM = <FS_UOM>-UNIT.

If nothing found in internal table ITB_UOM, check with source fields LOGSYS, COSTELMNT, U_UNITUM, C_PLANT and C_COSTCTR empty. If a correspondance is done and in function of the period: K_ACT_QTY * <FS_UOM>-K_UOMCVRT * <FS_UOM>-K_UMREZ / <FS_UOM>-K_UMREN and U_UNITUM = <FS_UOM>-UNIT.

If nothing found in internal table ITB_UOM, check with source fields LOGSYS, COSTELMNT, U_UNITUM, and C_PLANT + C_COSTCTR empties. If a correspondance is done and in function of the period: K_ACT_QTY * <FS_UOM>-K_UOMCVRT * <FS_UOM>-K_UMREZ / <FS_UOM>-K_UMREN and U_UNITUM = <FS_UOM>-UNIT.

Read internal table ITB_T006 and in function of the dimension, make a conversion of the quantities (examples [here](#)).

For dso ABECO202 and ABECO208

Production equipment (C_PRDEQUI) and Vecotr (C_VECTOR) are by default assigned to "ND" (not defined).

Read internal table ITB_PRDEQUI (from ZECR_AS_CC2EQ_T) with LOGSYS, COSTCTR and COSTELMNT if there is a correspondance in internal table, take the C_PRDEQUI from internal table. If no correspondance, search the same but with COSTELMNT empty, is correspondance take C_PRDEQUI.

If not, same principle with COSTELMNT and COSTCTR empties.

Read internal table ITB_VECTOR (from ZECR_AS_CE2VC_T) with LOGSYS, COSTCTR, C_MATNR2 and COSTELMNT if there is a correspondance in internal table, take the C_VECTOR from internal table. If no correspondance, search the same but with COSTCTR empty, is correspondance take C_VECTOR.

If not, same principle with C_MATNR2 empty, then C_COSTCTR + C_MATNR2 empties, then C_COSTCTR + C_MATNR2 + C_COSTELMNT empties.

For dso ABECO203 and ABECO211 same logic but with STKEYFIG instead COSTELMNT.

For dso ABECO201 and ABECO207 same logic but with C_HRKFT + C_INDEX1 + C_KSTRG4 + LOGSYS or LOGSYS + C_HRKFT + C_HRKFT + C_KSTRG4 + C_ORIMAP3.

After these steps if we have a C_PREDQUI and this one is present in internal table ITB_SITE (master data C_PRDEQUI).

Read internal table ITB_VFLAG (table ZECR_AS_VFLAG_T) in function of site of internal table ITB_SITE.

If a result is found and in function of fiscal period, if last two characters of C_VECTOR = SO C_PRCOIND = P. Else if (source fields DB_CR_IND = S and K_ACT_QTY >= 0) or two last characters of C_VECTOR = PU C_PRCOIND = PU. Else C_PRCOIND = P.

Further explanations for program ZBW_ZECO2_FILE - ZECO2_FILE Transaction

ECO2 - Loading of flat files



Choose the file to load

File to load

Choose your Excel File



Preview



Check file



Load file



History



Exit

User can choose an Excel flat file from his local computer by using the "File to load" browser.

[blocked URL](#)

Once the file is chosen, the user can preview the content to be uploaded from his selected file by clicking on "Preview" icon.

Preview is color-coded: green entries are correct, while red entries contain one or more errors based on Business Rules detailed below.

Once the file is chosen, the user can start the content checking of the file by clicking on "Check file" icon.

If errors exists, they will be listed with the associated error code and long text (MD C_REJECT) at the end of each line.

[blocked URL](#)

Once the file content is checked and all errors corrected, the user can load the file to BW application server (AL11) and raise the event associated with the type of import they chose by clicking on "Load file" icon.

The event raised Z_EVT_PC_CO_ECO2_22 triggers the process chain PC_CO_ECO2_22 which loads the user's file data into CO2 ADSO ABECO206.

Source Code

Report ZBW_ZECO2_FILE contains several distinct forms:

Interface

Selection screen.

Import File

f200-import_file: Import the XLS file from user's workstation.

f210-upload_excel: Convert the XLS file in Internal table.

f220-check_file: Check content of the file as follow:

- BUSINESS RULE 1. **DATASOURCE VALUE** - ECO2_BR1 (Rejection Code)

Authorized values: SERT and EPS

- BUSINESS RULE 2. **FISCAL PERIOD VALUE** - ECO2_BR2

Between 001.1900 and 012.9999

- BUSINESS RULE 3. **PRODUCTION EQUIPMENT EXISTS** - ECO2_BR3
- BUSINESS RULE 4. **VECTOR EXISTS** - ECO2_BR4
- BUSINESS RULE 5. **PROD/CONS INDICATOR VALUE** - ECO2_BR5

Authorized values: C (Consumption) and P (Production)

- BUSINESS RULE 6. **VALIDATION FLAG VALUE** - ECO2_BR6

Authorized values: 'X' or empty

- BUSINESS RULE 7. **UNITS EXIST**
 - BUSINESS RULE 7.1. **AI UNIT EXISTS** - ECO2_BR7_1
 - BUSINESS RULE 7.2. **STANDARD UNIT EXISTS** - ECO2_BR7_2
- BUSINESS RULE 8. **CURRENCY EXISTS** - ECO2_BR8
- BUSINESS RULE 9. **KEY FIGURES ARE NUMERICS** - ECO2_BR9
- BUSINESS RULE 10. **NO DUPLICATED ENTRIES** - ECO2_BR10

Program Output

f300-output_preview: Generate ALV grid to show Converted data from Excel.

f310-output_check: Generate ALV grid to show Converted data from Excel after check.

Program Load

f400-load_pchain: Check if process chain is already in progress.

f413-csv_file_data: Import Customer file in ALL11.

- Filename consumed by process chain PC_CO_ECO2_22 is hardcoded in this form: "/exploit/BW/ECO2/ECO2_flat_file.csv".
- Flat file integrated manually by user, is saved as follow: '/exploit/BW/ECO2/' w_file '_' sy-sysid '_' sy-uname '.csv'.
Where *w_file* is the filename on user's local pc, *sy-sysid* is the BW environment and *sy-uname* is the user name which is running the program.

f423-filename_data: Update Global filter master data with new file name.

f433-savelog_data: Update log after loading (ADSO APECO201 - "ECO2 - Flat Files Log").

f440-finish_message: Pop- up message once the program's operations are complete .

f453-send_email_data: Send an email after data loading.

Program Log

f500-display_log: Display log.

Historisation of Conversion Data

Based on Validation Flag set in ZECO2 for a given Site/Period.

ABECO204 ABECO205 ABECO212

Read ABECO212 & ABECO205 when loading ABECO204.

Load ABECO205 only with entries where Validation Flag = "

Before reloading ABECO204 & ABECO205, check ABECO205 entries and load every entry where Validation Flag = X in ZECO2 table into ABECO212.

Delete entries in ABECO212 if they exist in ABECO204 with Validation Flag = ".

Reporting

Query code	Query description	Details
BW_QRY_CPECO201_0000	ECO2 - DataSource for ABECO204 (Core Query)	Used only, as the name indicates, as DataSource for ADSO ABECO204. Reports all entries in CPECO201
BW_QRY_CPECO201_0001	ECO2 - SAP database before CO2 conv. (Core Query)	Reports entries from ADSOs ABECO201/02/03/07/08
BW_QRY_CPECO201_0002	ECO2 - Manual entries before CO2 conv. (Core Query)	Report entries from ADSO ABECO209
BW_QRY_CPECO202_0001	ECO2 - SAP database after CO2 conv. (Core Query)	Report entries from ADSOs ABECO205/10
BW_QRY_CPECO202_0002	ECO2 - Historical data from external file incl. CO2 conv. (Core Query)	Report entries from ADSO ABECO206
QV_BW_QRY_CPECO201_0001	ECO2 - QV interface before CO2 conv. (Core Query)	Qlikview query - Report all entries in CPECO201
QV_BW_QRY_CPECO202_0001	ECO2 - QV interface after CO2 conv. (Core Query)	Qlikview query - Report all entries in CPECO202

Dependencies with other applications

We should have the information where the application is sending or receiving information (e.g. APD open hub)

Data Loading

Info Providers and objects loaded

Data produced by CERISE is to be used in QlikView.

Data is collected from the following existing BW applications:

- IMEP Main Dataflow PF1 (APCOPP01) + WP1 (APCOPP02)
- COCA : Cost Center Accounting (DPCOCA01, ODS_CCA9 & ODS6CCA9)
- ODS_CCA9 data is enhanced through ADSO APCOCT01

Process Chain	Code	Type	Frequency	Comments
ECO2: TD - T - Cerise Transactional Data	PC_CO_ECO2_01	MAIN	Twice daily	Runs at 2:00 (CET, AM and PM)
ECO2: MD - D - Cerise Master Data	PC_CO_ECO2_02	MAIN	Daily	Runs at 3:30 AM CET

ECO2: MD - D - Cerise Master Data - Attributes	PC_CO_ECO2_03	SUB (02)	Daily	
ECO2: MD - D - Cerise Master Data - Texts	PC_CO_ECO2_04	SUB (02)	Daily	
ECO2: MD - T - Cerise Master Data - Vector Category	PC_CO_ECO2_05	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Vector Type	PC_CO_ECO2_06	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Vector Class	PC_CO_ECO2_07	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Vector Group	PC_CO_ECO2_08	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Prod. Unit Category	PC_CO_ECO2_09	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Prod. Unit Class	PC_CO_ECO2_10	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Prod. Unit Group	PC_CO_ECO2_11	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Production Unit	PC_CO_ECO2_12	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Vector	PC_CO_ECO2_13	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Prod. Equipement	PC_CO_ECO2_14	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - ETS Register	PC_CO_ECO2_15	MAIN	Trigger	
ECO2: MD - T - Cerise Master Data - Site	PC_CO_ECO2_16	MAIN	Trigger	
ECO2: TD - T - Cerise Transactional Data - WP1	PC_CO_ECO2_17	SUB (01)	Twice daily	
ECO2: TD - T - Cerise Transactional Data - Qties/Costs Entry	PC_CO_ECO2_19	MAIN	Trigger	
ECO2: TD - T - Cerise Transactional Data - Data Entry	PC_CO_ECO2_20	MAIN	Trigger	
ECO2: TD - T - Cerise Transactional Data - CO2 conversion	PC_CO_ECO2_21	SUB (01)	Twice daily	
ECO2: TD - T - Cerise Flat Files Import	PC_CO_ECO2_22	MAIN	Trigger	

Loading frequency

Master Data (PC_CO_ECO2_02) loaded daily + on-demand upon ZECO2 Transaction run.

Transactional Data (PC_CO_ECO2_01) loaded twice daily.

Flat Files import (PC_CO_ECO2_22) on-demand.

Average performance

Key Figure	Estimation
~ Average Process Chain Runtime	PC_CO_ECO2_01: 5min PC_CO_ECO2_02: 1min PC_CO_ECO2_05 to 20: <1min PC_CO_ECO2_22: <1min
~ Average nb of rows loaded per load	ABECO201: 180k ABECO202: 3500 ABECO203: 19k ABECO204: 16k ABECO205: 16k ABECO207: 250k ABECO208: 3000 ABECO209: <100 ABECO210: <100
~ Average Runtime for 10k lines	~1min for most chains

Record Keeping

All data is deleted before each load, no history to speak of outside of ABECO206 (Historical data from previous storage) which currently contains data up to 6 years prior (2015).

Global filters (C_GLBFLT) used in Chains

Stream	Rule	Counter	Explanation
ECO2	FF_DATA	1	Input filename for ZBW_ZECO2_FILE program. Used to save a csv copy in AL11 before proceeding with Flat File Import.
		2	~~Store actual data queue of running process chain~~ Deprecated
		3	Total counter for ZBW_ZECO2_FILE program file loading. Used for easy access to the last X entries in logs ADSO APECO201.

ABAP Programs used for Data Load

Program	Variants	Explanation
ZECO2_LOAD_VIEWS	-	Dedicated program associated with transaction ZECO2 to handle Manual update of Filter, Assignment and Conversion tables as well as Flat File import.
ZBW_ZECO2_FILE	-	Dedicated program created to allow users to load flat files to integrate historical Cerise data. Associated with transaction ZECO2_FILE. Program is not used in Process Chain, but launches event PC_CO_ECO2_22 starting PC of the same name.

Further explanations for program ZECO2_LOAD_VIEWS - ZECO2 Transaction - ZECR Function Group

2) SAP data filtering

- ☐ Periods
- ☐ Plants
- ☐ Cost Centers
- ☐ Cost Elements

Filter tables	SE11 Table (_T) & View (_V)	ZECR Screen (SE80)	Explanation
Periods	ZECR_FT_PER_T	23	Set values for OLAP variables V_0FISCPER_0014 (Solvay) and V_0FISCPER_0015 (Rhodia)
Plants	ZECR_FT_PLT_T	5	Set values for OLAP variables V_C_PLANT_0018 (Solvay) and V_C_PLANT_0019 (Rhodia)
~~Origin groups (PF3)~~	~~ZECR_FT_OGP_T~~	~~24~~	~~Set values for OLAP variables V_C_HRKFT_0001 (Solvay) and V_C_HRKFT_0002 (Rhodia)~~
Cost Centers	ZECR_FT_CC_T	25	Set values for OLAP variables V_C_FUNCT_3_0004 & V_C_KOSAR_0002 (Solvay) and V_C_FUNCT_3_0005 & V_C_KOSAR_0003 (Rhodia)
Cost Elements	ZECR_FT_CE_T	26	Set values for OLAP variables V_C_CELTHX2_0001 (Solvay) and V_C_CELTHX2_0002 (Rhodia)
~~SKFs (PF3)~~	~~ZECR_FT_SKF_T~~	~~27~~	~~Set values for OLAP variables V_C_SKFHEC1_0001 (Solvay)~~

1) Master data definition

- ☐ Site
- ☐ ETS Register
- ☐ Prod. Unit Category
- ☐ Prod. Unit Class
- ☐ Prod. Unit Group
- ☐ Production Unit
- ☐ Prod. Equipment (PEQ)
- ☐ Vector Category
- ☐ Vector Type
- ☐ Vector Class
- ☐ Vector Group
- ☐ Vector (VCT)

Definition tables	SE11 Table (_T) & View (_V)	ZECR Screen (SE80)	Explanation
Vector Category	ZECR_DF_VCTCAT_T	12	Fill Master Data Attributes and Texts of C_VECTCAT. Runs PC_CO_ECO2_05
Vector Type	ZECR_DF_VCTTYP_T	13	Fill Master Data Attributes and Texts of C_VECTTYP. Runs PC_CO_ECO2_06
Vector Class	ZECR_DF_VCTCLS_T	14	Fill Master Data Attributes and Texts of C_VECTCLA. Runs PC_CO_ECO2_07
Vector Group	ZECR_DF_VCTGRP_T	20	Fill Master Data Attributes and Texts of C_VECTGRP. Runs PC_CO_ECO2_08
Vector (VCT)	ZECR_DF_VCT_T	21	Fill Master Data Attributes and Texts of C_VECTOR. Runs PC_CO_ECO2_13
Prod. Unit Category	ZECR_DF_PRDCAT_T	15	Fill Master Data Attributes and Texts of C_PRDCAT. Runs PC_CO_ECO2_09
Prod. Unit Class	ZECR_DF_PRDCLS_T	16	Fill Master Data Attributes and Texts of C_PRDCLA. Runs PC_CO_ECO2_10
Prod. Unit Group	ZECR_DF_PRDGRP_T	17	Fill Master Data Attributes and Texts of C_PRDGRP. Runs PC_CO_ECO2_11
Production Unit	ZECR_DF_PRU_T	18	Fill Master Data Attributes and Texts of C_PRDUNIT. Runs PC_CO_ECO2_12
Prod. Equipment (PEQ)	ZECR_DF_PEQ_T	19	Fill Master Data Attributes and Texts of C_PRDEQUI. Runs PC_CO_ECO2_14
ETS Register	ZECR_DF_REG_T	29	Fill Master Data Attributes and Texts of C_ESTREGI. Runs PC_CO_ECO2_15
Site	ZECR_DF_SIT_T	22	Fill Master Data Attributes and Texts of C_ZZRSIT. Runs PC_CO_ECO2_16

When leaving the Screen after editing a table, an event is launched to run the associated Process Chain. **Event Z_EVT_[PC_NAME] runs [PC_NAME]**.

4) SAP data assignment

- ☐ Work Ctr to PEQ (WP1)
- ☐ Cost Obj to PEQ (PF1)
- ☐ Cost Ctr to PEQ (PF1)
- ☐ Ind1 Work Ctr to VCT (WP1)
- ☐ Ind3 Material to VCT (WP1)
- ☐ Ind5 Cost Elt to VCT (WP1)
- ☐ Ind1 Cost Obj to VCT (PF1)
- ☐ Ind3 Origin Grp to VCT (PF1)
- ☐ CC Cost Elt to VCT
- ☐ SKF to VCT (PF1)

Assignment tables	SE11 Table (.T) & View (.V)	ZECR Screen (SE80)	Explanation
Ind1 Work Ctr to VCT (WP1)	ZECR_AS_PL2VC_T	33	Force C_VECTOR value for key 0LOGSYS/C_PLANT/C_WORKCTR/C_MATNR2
Ind1 Cost Obj to VCT (PF1)	ZECR_AS_CO2VC_T	6	Force C_VECTOR value for key 0LOGSYS/C_KSTRG4/C_INDEX1
Ind3 Material to VCT (WP1)	ZECR_AS_OM2VC_T	34	Force C_VECTOR value for key 0LOGSYS/C_ORIMAT/C_PLANT/C_WORKCTR
Ind3 Origin Grp to VCT (PF1)	ZECR_AS_OG2VC_T	7	Force C_VECTOR value for key 0LOGSYS/C_HRKFT/C_PLANT/C_KSTRG4/C_ORIMAT
Ind5 Cost Elt to VCT (WP1)	ZECR_AS_CE2VC2_T	35	Force C_VECTOR value for key 0LOGSYS/0COSTELMNT/C_PLANT/C_WORKCTR
CC Cost Elt to VCT	ZECR_AS_CE2VC_T	8	Force C_VECTOR value for key 0LOGSYS/0COSTLMNT/C_COSTCTR/C_MATNR2
SKF to VCT (PF1)	ZECR_AS_SK2VC_T	9	Force C_VECTOR value for key 0LOGSYS/0STKEYFIG/C_COSTCTR
Work Ctr to PEQ (WP1)	ZECR_AS_PL2EQ_T	32	Force C_PRDEQUI value for key 0LOGSYS/C_PLANT/C_WORKCTR/C_MATNR2
Cost Obj to PEQ (PF1)	ZECR_AS_CO2EQ_T	10	Force C_PRDEQUI value for key 0LOGSYS/C_KSTRG4
Cost Ctr to PEQ (PF1)	ZECR_AS_CC2EQ_T	11	Force C_PRDEQUI value for key 0LOGSYS/C_COSTCTR/0COSTELMNT

In all assignment tables, an empty key field means "ALL VALUES". Keys are listed in order of priority to determine assignment value.

Ex. for ZECR_AS_PL2VC_T: If no match is found for key 0LOGSYS/C_PLANT/C_WORKCTR/C_MATNR2, check key 0LOGSYS/C_PLANT/C_WORKCTR/, then 0LOGSYS/C_PLANT/"", then 0LOGSYS/""/". If still no match is found, assignment is set to default value ('ND' for C_VECTOR and C_PRDEQUI).

If the table forces C_VECTOR, leaving the Screen releases event Z_EVT_PC_CO_ECO2_13 (which runs PC_CO_ECO2_13). If it forces C_PRDEQUI, it releases event Z_EVT_PC_CO_ECO2_14 (which runs PC_CO_ECO2_14).

3) SAP data update: exception management

- ☐ Forced UoM for Cost Elt qty
- ☐ Forced UoM for SKF qty (PF1)
- ☐ Fake Prod Line conv. (WP1)

Exception management tables	SE11 Table (.T) & View (.V)	ZECR Screen (SE80)	Explanation
Forced UoM for Cost Elt qty	ZECR_CV_UOM2_T	31	Force conversion rate and target unit for key 0LOGSYS/0COSTELMNT/U_UNITUM/C_PLANT/C_COSTCTR and From/To Periods. C_PLANT and C_COSTCTR can be empty, everything else is mandatory.
Forced UoM for SKF qty (PF1)	ZECR_CV_UOM_T	28	Force conversion rate and target unit for key 0LOGSYS/0STKEYFIG/U_UNITUM/C_PLANT/C_COSTCTR and From/To Periods. C_PLANT and C_COSTCTR can be empty, everything else is mandatory.
Fake Prod Line conv. (WP1)	ZECR_CV_PRL_T	36	Default rule for ABECO207 is to collect Production Line C_MDV01 first 8 characters as Resource C_WORKCTR, but it does not work for some Production Lines. Force Resource value for key 0LOGSYS/C_MDV01/C_PLANT.

5) Recurring operations

- ☐ Energy&CO2 conv. rates entry
- ☐ Qties/Costs manual entry
- ☐ CO2 Process manual entry
- ☐ Content transfer btw Sites

Recurring operations	SE11 Table (_T) & View (_V)	ZECR Screen (SE80)	Explanation
Energy&CO2 conv. rates entry	ZECR_AS_CO2_T	30	Force ratios for key C_VECTOR/C_ZZRCST/C_PRDEQUI and From/To Periods.
CO2 Process manual entry	ZECR_EN_EMPR_T	38	Manual entry of data to assign K_SC1EMPR and K_ETSEMPR to key C_PRDEQUI/C_VECTOR /C_PRCOIND and From/To Periods. Data is saved in ABECO210 to be matched with ABECO205 data in Composite Provider CPECO202.
Qties/Costs manual entry	ZECR_EN_ACTU_T	39	Manual entry of data to assign Costs and Quantities to key C_PRDEQUI/C_VECTOR/C_PRCOIND and From /To Periods. Data is saved in ABECO209 and accessible through Composite Provider CPECO201.
Content transfer btw Sites	ZECR_AS_TRSF_T	24	Reassign data from one Site to another, based on C_PRDEQUI and C_VECTOR

6) Non-recurring operations

- ☐ Data upload from ext. file
- ☐ Validation Flag by Site/Period

Non-recurring operations	SE11 Table (_T) & View (_V)	ZECR Screen (SE80)	Explanation
Data upload from ext. file	-	-	Runs program ZBW_ZECO2_FILE
Validation Flag by Site/Period	ZECR_AS_VFLAG_T	37	Force C_FVALID value for key C_ZZRCST with From/To Periods

Data Quality Control

Operational Documentation

Procedures

<Describe the recurring procedures needed to operate the application (eg. start/pause/terminate/restart the app processes, data preparation, data ingestion, ETL, data visualization, data export, other manual activities)>

Scheduling

<Describe the scheduling in place for the application (eg. existing jobs, trigger time/event based, dependencies)>

Monitoring

<Describe the monitoring checks to confirm the application is performing well (eg. check the overall status, check performance metrics like runtime/data volume/memory/disk/CPU, maintain and react to alerts/notifications)>

Error Handling

<Describe how to handle errors (eg. error codes, description and respective resolution, alert users)>

Known Bugs

<List the existing bugs, its criticality, workarounds and resolution plan.>

Roadmap

<List past & future evolutions for the application (including links to MED/FSD/TSD)>