KERN

www.kern.ag

Allevo ProCED

User manual

Valid from Allevo 4.3

ProCED - Provide Cost Element Dynamics

In many cases, SAP objects cannot be planned according to a fixed cost element structure, since very different planning processes are used depending on the business-specific use case. Why should there be cost elements in a planning form that are unnecessary in 90% or more of individual planning cases, but which decrease performance because of constant communication with the system?

Allevo ProCED aims to determine the optimal and regular individual structure of a cost element for the respective planning object in advance, and to create the planning form in a dynamic process at run-time. To achieve this, it is necessary to specify certain rules for the relevant planning structure of an object or an object type. These are described in this manual.

Content

Table of Contents

1 Allevo ProCED - Introduction

ProCED is particularly useful when planning is done in a list-oriented manner using Allevo Multi-Object. In this case, ProCED determines all planning-relevant information about a group of objects (e.g. cost center group or project) and stores the data in a satellite table using a so-called "representative object": normally, this is the first planning object of a group or project. For this representative object, information that does not belong directly to the selected object type are also stored, if necessary (e.g., data relating to internal orders when entering via a cost center).

Entry into Allevo planning transactions also takes place via group or project number.

Note:	The Allevo functionality of MOD, which is described in the Allevo Excel manual, follows another approach similar to that of ProCED. The goal of MOD is to be able to freely select any objects and cost elements in the start screen of Allevo. The planning form then generates a structure as a combination of these objects and cost elements, taking into account rows for subtotals. Optionally, additional data for MOD-mode can also be provided via ProCED. Whether a representative object can be used is therefore also an important criteria in deciding which variant to choose for an Allevo implementation project.

1.1 Basic Functions

If a planning form is to be generated dynamically by the planning system, the system must know how the form structure is set up.

ProCED takes three approaches to this:

1. In principle, the structure is based on the existing posting relationships for the object. From this, the relevant cost elements and their further planning information can be derived. With the analysis of these past data, restrictions can be made with respect to cost elements, value types, and transactions.
2. Additionally, cost elements or statistical key figures which should be included in the dynamic form can be specified, regardless of whether they have been used in the past.

However, a fundamental problem now arises whenever, for example, there are no historical posting relationships for new objects, and no planning-relevant cost elements can be supplemented by the second approach. In this case, it is possible to:

3. Add additional cost elements manually and individually in the generated Excel planning form.

1.2 Determine Posting Relationships for Objects

The ProCED module determines the planning-relevant details of existing posting relationships for cost centers, orders, profit centers and projects (WBS elements) and enters these into a freely definable satellite table. This includes

- Object (number and text)

- Cost elements with cost element type or statistical key figures
- Partner and partner activity type
- Hierarchy information (for calling via groups)

Here we are only interested in the initial provision of the relevant posting relationships. ProCED therefore places these structures in the respective satellite table, where they can be entered into the Allevo master when Allevo planning is called.

The actual reading of the data is done with the normal reading process of Allevo.

Note:	Using ProCED makes sense in the context of list-oriented planning via Allevo-MOM, but it can also be used in Multipage-planning.

1.3 Starting ProCED

ProCED can be started in three ways:

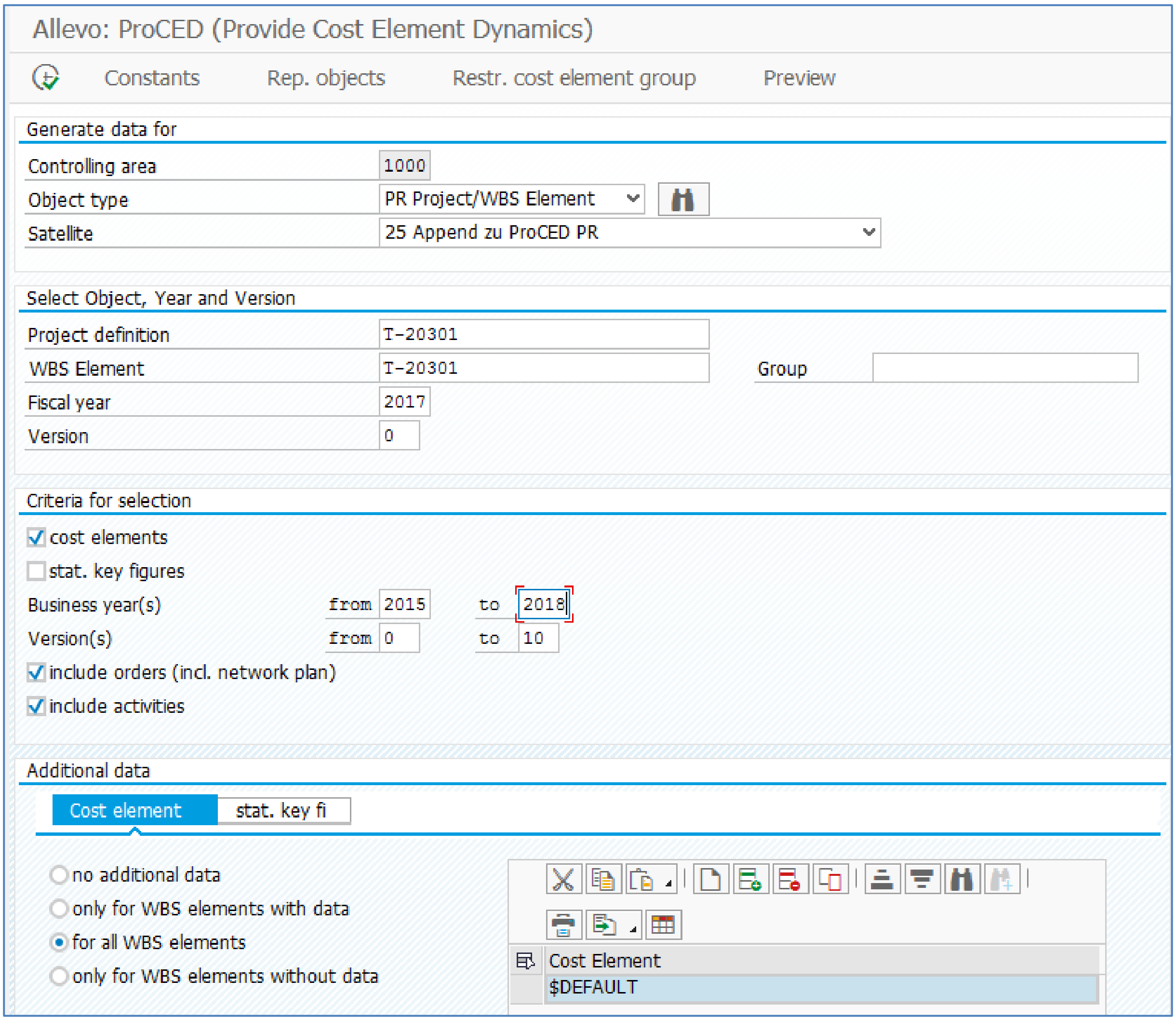
1. Via dialog transaction /KERN/IPPPROCED, all steps for setting up ProCED are carried out. As in the Allevo main module, many functions are controlled via constants: some of these constants define the field mapping in order to correctly find the desired data in the satellite table (see description of the constants in section **Error! Reference source not found.**). In execution mode, data are directly entered into an object or group in the selected satellite table (see next section).
2. In addition, there are the report transactions /ALLEVO/PROCED_REP only used for execution: Using a predefined selection, data are written in a satellite table and are then available for Allevo planning. The selection can take place via any number of objects (also with multiple selection) or a group. In order to simplify recurring selections, variants can be saved.
3. ProCED can also be called directly from Allevo: in this case, data selection is carried out for the specific object selected in Allevo (see section 3.4).

The execution methods described in points (2) and (3) take into account all of the settings made in (1) for the selected object type and satellite table.

1.4 Set Up ProCED Functions

1.4.1 Call Transaction /ALLEVO/PROCED

Allevo ProCED is started and set up via transaction /ALLEVO/PROCED (see Figure 1.1).



Allevo: ProCED (Provide Cost Element Dynamics)

Generate data for

Controlling area	1000
Object type	PR Project/WBS Element
Satellite	25 Append zu ProCED PR

Select Object, Year and Version

Project definition	T-20301		
WBS Element	T-20301	Group	
Fiscal year	2017		
Version	0		

Criteria for selection

☑ cost elements
□ stat. key figures

Business year(s)	from	2015	to	2018
Version(s)	from	0	to	10

☑ include orders (incl. network plan)
☑ include activities

Additional data

○ no additional data
○ only for WBS elements with data
● for all WBS elements
○ only for WBS elements without data

Cost Element
$DEFAULT

Figure 1.1: Starting Allevo ProCED (shown for object type "PR")

Here you first have to define the object type to work with (cost center, order, project/PSP, profit center or business process) and in which satellite table the results should be stored. The selection list for satellites contains only those satellites that have been activated. As an additional search function, a list of all satellites that have been set up for ProCED in the selected controlling area (i.e. where a scheme with constants is already saved) can be called.

For the execution of the program, the relevant object must be entered as a single object or as a group: In order to determine the posting relationships, a group is always completely dissolved into all of its individual elements. Dynamic groups are also allowed: These so-called "selection variants" are created via transaction KM1V (cost center) or OKOV (orders). As usual, the entry in the input field of the selection mask must begin with a period (e.g. ".TEST").

Note:	It is also possible to use "mixed" groups, which are provided by the SAP standard transactions for groups (e.g., KAH2 for cost center groups). There, selection variations can be contained as subordinate groups, which Allevo may dissolve dynamically.

The result of the selection (list of posting relations) is taken over by ProCED into the entered satellite. The button [Preview] allows to display the satellite data without real saving (especially useful for testing).

1.4.2 Special Features for Object Type PR

For object type PR it is also possible to enter a project number or a WBS element as an alternative to the group. The following rules apply for the treatment in ProCED:

- A corresponding representative WBS element is automatically determined for the project and displayed on the initial screen. ProCED takes into account the posting combinations of all WBS elements in the project and stores them for the representative object in the satellite.

- When entering a single WBS element, all WBS elements contained in the corresponding part of the project hierarchy are taken into account (the rule applies at least to current program versions).

- When entering a group, it is only the PSP elements that are included in this group (without subordinate hierarchy).

An equivalent logic is also available if ProCED is dynamically integrated in Allevo main planning (see also individual definition via the constant PSP_MULTI).

1.4.3 Additional Criteria for Data Selection

In the screen area "Criteria of data selection" is defined for which (historical) accounting relationships ProCED should search: Cost elements and "statistical key figures" are available for selection. The specification of fiscal years and versions define the time range in which ProCED should search.

Note:	When reading from the source object, it is important that the matching KL objects are also included in the selection. For this purpose, the selection of the version from the initial screen is supplemented by the "version for indirect activity allocation" from Customizing for the fiscal year (TKA07-VPDIO).

In the initial screen, other options such as "Include Orders (incl. Network plan)" and " Include Business Transactions " are also only relevant for object type PR: The selection then additionally includes all orders or network activities that are assigned to a WBS element via the respective master record (i.e. table AUFK and AFVC): When transferred to the satellite, additional rows are generated if necessary (with specification of the respective object type, e.g. NP).

Note:	Allevo stores the mentioned criteria for data selection in the contant SEL_CRITERIA. Optionally another criterion allows there also the access to ACOSX tables (see also F1 help there).
	If desired, the found posting relationships can be summarized at the group level before being transferred to the satellite table (constant NODE_AGGREGATION).

1.4.4 Additional Data

Entries in this range allow the result to be extended to posting relationships from historical relationships, with respect to defined cost elements and statistical key figures (as an additional fixed part of the result). The transfer to the result list can be carried out selectively for all selected objects, those without data or with data (depending on the use case).

Note:	This function can be useful in order to have at least one entry in the result list for each object.

The procedure for the execution of ProCED is described in the following on the basis of cost elements (but also applies for statistical key figures).

If the list contains several entries in the "cost element" category, ProCED transfers these entries to the appropriate object in the results list (according to specified parameters, for example "for all WBS elements" as in the figure above). This can also include dummy cost elements: In this case there is no alignment with the saved master records.

If only a single cost element is entered, ProCED also checks for the cost element group with the following steps:

- A cost element is directly transferred if a valid master record is saved.

- If it is not a valid cost element, ProCED tries to find a cost element group with this name in the system or the group is dissolved in order to transfer all cost elements from the result list, if necessary.

- If neither of these cases apply, ProCED transfers the entry as if it were a dummy cost element (without checking for a master record).

This same logic applies to statistical key figures. The entries in the list are saved in the constant DEFAULT_KSTAR or DEFAULT_STAGR (for details see F1 documentation or section **Error! Reference source not found.**).

As an alternative to the process described here, a restriction to the cost element group can also be made without the start object (see section **Error! Reference source not found.**).

1.4.5 Additional Functions

Additional functions for setting up the interface can be called up via the toolbar:

- Via |Constants| the assignment of the satellite fields to the relevant program parameters of ProCED is done. The setup of these constants will always be the first step when setting up ProCED (see section 5).

- Via button |Rep.objects| the representative objects are stored, which are used for the application of ProCED in connection with Allevo-MO (see section 2.1).

- For the selection of historical postings it can be useful to limit the list of relevant cost elements (or accounts for object type PC) from the beginning; this can be done by clicking on the button |Restr. cost element group| (see section 4.1).

- The preview function is already mentioned in a previous section.

These additional functions can only be called up if the object type and the satellite table have been entered beforehand.

1.5 GroundTables

As of Allevo version 3.5.17, the Allevo delivery package includes several GroundTables that can be used to easily map standard cases for working with ProCED (use cases).

Table name	Description
/CORE/U_PRCD01_A /CORE/U_PRCD03_A	Data provision via ProCED for object type OR or PR; the tables are duplicated (i.e. /KERN/U_PRCD0_A2), as they are often required twice in Allevo implementation projects.

For the basic function of GroundTables in connection with Allevo satellites, see the Allevo SAP manual.

2 Variants of the Execution of ProCED

For Allevo ProCED, there are different execution variations that are described in more detail in the following sections:

- ProCED and Allevo-MO for a list-oriented application (section 2.1)
- ProCED and MultiPage mode (section 3.1)
- ProCED with call MOWO for variable sheet assignment (section 3.2)
- ProCED and MOD-mode (section 3.3)

Note: The order of the reading processes of Allevo is important when used in connection with ProCED. If a satellite contains information about cost elements, this must be read from Allevo before the general reading of the cost element data from the form can be carried out.

When using satellites 01 to 99, pay attention to the constant READ_ORDER_SAT; the sequence is determined by the appropriate entry in the field "Value from" in the constant. To do this, use the assistent in the constant maintenance (ROS column).

2.1 List-oriented Planning with Allevo-MO (MultiObject Mode)

ProCED is particularly useful when planning is done in a list-oriented manner using Allevo MultiObject. In this case, all data of a particular object group (i.e. cost center group or project) are saved in the satellite table under a representative object or group.

Satellite table 0 is particularly useful for this variant, however other satellites can also be used here.

2.1.1 Determining the representative object for a group (projects)

With this variant, the data are selected after entering a project. Alternatively, different WBS elements can be combined in a group.

Calling via a project:

ProCED runs through an extended determination logic for the representative object as soon as it is started via the specification of a project number. This ensures that ProCED saves its cost element details under the same representative element in the satellite table, just as it is required in the Allevo main program. Depending on the system settings, this can be the first WBS element or the first element that has a planning indicator. The following settings are available in the ProCED constant REP_OBJECT:

Value from: entry of the level from which the WBS element is selected
empty = first element according to "Value to”

Value to: empty = first element (with or without planning indicator)

1 = first element **without** planning indicator

2 = first element **with** planning indicator

3 = Standard determination like Allevo main program (with planning indicator)

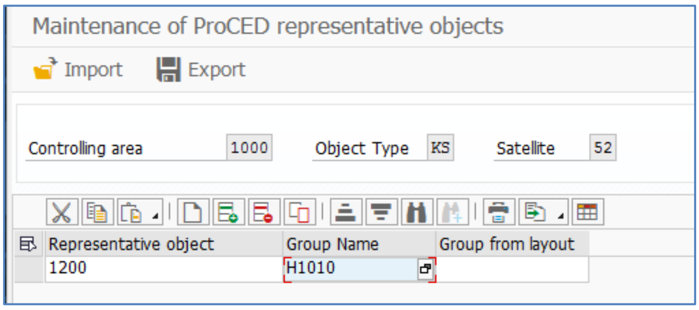
Note:	The settings in the constant REP_OBJECT are only relevant if a PROJECT is specified in the ProCED initial dialog or when called from the Allevo main program. Only then is the representative object determined according to the specifications. If, in addition to the project, a WBS element is also specified (as required for a group), this WBS element is used as a representative object.

Calling via WBS groups:

When ProCED is called via a WBS group, a two-level determination is run for the representative object. The first step is to search for a valid entry in the 1/n table (see section below). If there is no entry, ProCED uses the entry in the "WBS element" field (an entry must be made here if necessary).

2.1.2 Determination of the Representative Object for the Group

If ProCED is called from the Allevo main program for cost centers, orders or WBS elements, it must be determined whether the individual object submitted by the main program stands alone or is representative of a group. Therefore, a table view is available in ProCED – depending on the satellite table and object type (button |Rep. Objects|):



Maintenance of ProCED representative objects

Controlling area 1000 Object Type KS Satellite 52

Representative object	Group Name	Group from layout
1200	H1010	

Figure 2.1: Allocation of a group to a representative object

If the transferred object is contained in this table as a representative object, the group is used for the data search. However, the data are still saved under the representative object in the satellite table. If the transferred object is not entered, only this object is selected and the results are saved only for this object.

Alternatively, the group can also be determined via the 1/n group settings of the Allevo main program. In this case, the corresponding layout must be entered in the field "Group from layout" (read from table /KERN/IPPCOOBJGR). If nothing is found for the entered layout, ProCED also reads entries in the * Layout.

2.2 Cross-object Type Read (READ_ASSIGNED_xx)

2.2.1 Basic Functions

In the normal case, ProCED reads data for the object type that was selected at the beginning of the transaction (or when called from Allevo main module). Alternatively, however, posting relationships of assigned objects of another object type can also be read. Example: Call ProCED via a cost center and determine all posting relationships for internal orders assigned to the cost center. Currently, the following combinations are possible:

- Entry via KS with Read posting relationships for OR and PR (WBS elements)
- Entry via PC with Read posting relationships for OR, KS and PR
- Entry via PR (WBS) with Read posting relationships for OR

The corresponding constants are READ_ASSIGNED_OR, READ_ASSIGNED_KS and READ_ASSIGNED_PR.

In the first program versions, data was **only** ever read for the assigned object type (e.g. assigned orders); but not directly for the initial object (e.g. cost center or profit center): Determining posting relationships for both object types in a list is only possible as of Allevo 4.0 (see notes in the F1 documentation on READ_ASSIGNED_OR).

It has been possible for some time to make a combined selection using different object types and storing the posting relationships in the same satellite (e.g. orders and WBS elements for the same cost center): For this, several constants may have to be active at the same time.

Note:	With entry via a cost center, the selection of the order/WBS data can also be made via the Profit Center to which the cost center is assigned. The function is also controlled via the constants READ_ASSIGNED_OR and READ_ASSIGNED_PR.

The constants mentioned must be stored in the scheme for the object type via which the access takes place (e.g. cost center/profit center): In addition, ALLEVO_ACTIVE must also be set there. The mapping of the transfer fields is always read from the scheme of the relevant object type: Normally, therefore, from assigned object types; if something is read in addition to the start object, the relevant satellite fields must also be assigned there (of course, each to the same satellite).

Note:	The found objects along with their posting relationships are finally stored in the satellite at the end with the ID of the initial object (as a representative object). Calling Allevo-MultiObject can still be done via the same start object. Sorting the list can be changed to a certain degree via the constant SORT_BY_KSTAR (see the corresponding F1 documentation for the constant)

The selection criteria that ProCED provides for the determination of assigned orders, cost centers, and WBS elements can be custom-defined using SAP selection variants. All criteria entered there are then taken into account in ProCED; only the characteristic for the start object (e.g., the cost center) is replaced by the current value when ProCED is called. This selection variant is saved in the field "Value to" of the constant READ_ASSIGNED_xx.

Important: A separate selection variant should be used for each scheme (exception PR, see below); when changing parameters with READ_ASSIGNED_xx, the selection variant must be manually checked (if necessary removing any required characteristics of the selection).

Note:	The use of SAP selection variants allows for a very flexible selection (e.g. also via the status of an order). Such flexibility comes at a price: The execution of these ProCED functions can also cost performance depending on the selected parameters and the amount of data in the customer system. The functions should therefore always be tested under real conditions.

Mixed planning of start object and assigned objects is possible from Allevo 4.0: Alternatively (and sometimes advantageously), two satellites can also be used in a MultiObject-master with individual data provision for each satellite. Another alternative: The posting elements for the entry object are entered permanently in the Allevo master.

Combination with MOD

If the functions for MOD are active in the Allevo MultiObject (e.g. constant DYN_KSTAR_SATxx), then Allevo allows access via several objects or a group.

If in this case also ProCED with object type-spreading reading is still merged, then it is of importance how the relevant start objects are passed to ProCED:

- In earlier Allevo versions, the group was passed directly and then resolved in ProCED into associated objects (as a basis for selection via READ_ASSIGNED_xx).
- As of Allevo 4.0, a group (or range) is resolved by Allevo and additionally matched with the current Allevo status: This filter then also applies when reading a subordinate object type.

The following applies in all cases: Posting relationships are always read completely for all start objects at the same time. Sorting in this "long" list is therefore primarily by object type.

MultiPage Mode

A direct call of ProCED in MP mode is normally done individually per sheet: Then also the object relevant there is transferred and ProCED determines the assigned objects of the deviating object type by that.

If the call is not made directly from Allevo but a list of posting relationships is created in advance via the dialog transaction, the constant MULTIPAGE_MODE must be set.

Call ProCED via 1:N Group

In the standard case, ProCED is called from Allevo via the current / representative object. Via the Allevo constant ONETON_PROCED, the group which is saved with the Allevo 1:N function can be optionally transferred.

Note:	Depending on the size of the 1:N group, it may be necessary to determine many posting relationships in ProCED and transfer these to Allevo. This can of course negatively impact the performance.

Authorizations and Status

For planning in Allevo, authorizations for all contained object types must be available (regardless of the start transaction). In individual cases, you can even plan without the start object: the start object (for example, a cost center) serves only as a framework for the determination of assigned objects (e.g., orders and WBS elements). In this case, a plan authorization must also exist for the initial object so that Allevo can open the planning mask (Allevo status must also be maintained).

WBS with Consideration of Assigned Orders

This is a different logic for reading based on the Allevo constant PSP_WITH_OBJECTS_OR (see details below).

2.2.2 Reading Order Data via Cost Center / Profit Center or WBS Element

The following applies to the constant READ_ASSIGNED_OR:

- **Value from** defines the organizational attribute in the order master to be used in evaluation (e.g., responsible or requesting cost center, Profit-Center of the cost center etc.).
- **Value to** contains the name of the selection variant with which the data is to be read (created using transaction OKOV). If required, several selection variants can be entered in "Value to" (comma-separated).

Background:	A selection variant remains blocked for the respective user until the corresponding master data has been determined. Another user must wait if necessary. If, however, there are several entries, Allevo automatically searches for the first free selection variant.

Further details can be found in the F1 help for the constant.

2.2.3 Reading Data for WBS Elements via a Cost Center / Profit Center

The following applies to the constant READ_ASSIGNED_PR:

Value from: Here, you define which organizational allocation is to be evaluated in the WBS master record.

Value to: The name of a selection variant with which the data is read can be optionally entered here (created using transaction CJ8V). The entry in the constant is done without the usual period. Without an entry, Allevo selects independently via table PRPS and the appropriate master record fields (see details in the documentation on the constant).

Note	Currently, only the fields of the PRPS table are taken into account when these WBS selection variants are resolved.

In addition to or instead of a selection variant, any WHERE condition can also be entered in **Value to** for the selection. Further details can be found in the F1 help for the constant.

The determined list of posting relationships is explicitly sorted by the number of the WBS elements.

2.2.4 Reading Cost Center Data via Profit Center

The following applies to the constant READ_ASSIGNED_KS:

Value from: The allocation of the cost center is defined here.

Value to: Here, enter the name of the selection variant with which the data are to be read (created via transaction KM1V). If required, several selection variants can also be entered in "Value to" (comma-separated).

Background:	A selection variant remains blocked for the respective user until the corresponding master data has been determined. Another user must wait if necessary. If, however, there are several entries, Allevo automatically searches for the first free selection variant.

Further details can be found in the F1 help on the constant.

2.3 WBS with Consideration of Assigned Orders

Background: This is a different function based on the Allevo constant PSP_WITH_OBJECTS_OR. A function is stored there to read costs of subordinate orders with a WBS element and then aggregate them to the WBS (see F1 documentation for the Allevo constant).

If ProCED is upstream, however, the associated Allevo WBS row must also be available, especially in the case where ONLY the associated orders bear costs. This function is provided by the ProCED constant PSP_WITH_OBJECTS_OR of the same name: It is only relevant when called up via object type PR and takes into account all orders that are assigned to the current WBS element (including networks). A limitation of the relevant orders is possible via order category and order type (for further details on the setup see F1-Document).

In contrast to the functions based on READ_ASSIGNED_XX described above, the constant here does not generate rows for the object type OR.

2.4 Reading Partner Information

Depending on the General constants for partner information (e.g. FD_PAROB, see section 5.2), Allevo also provides information on the partner or source object. If desired, ProCED generates a row in the transfer list for each partner:

- For the field PAROB, ProCED considers only KL objects (fields FD_PAROB, FD_PAROB_KOST, FD_PAROB_LSTA).
- There are no restrictions for the source object USPOB: One row is created for each source object as soon as there is a mapping (i.e. fields FD_UOBJECT, FD_UOBJTYPE...).

Depending on the process, however, you may want to carry out planning at a higher level and not via individual rows for each partner. In this case the constant SHOW_PARTNER can be helpful, which allows a filter e.g. via cost element type.

The fields for the source object are also relevant if posting relationships are read from the sender perspective, as described in the following section.

In the same way, the information about the receiver object (field FD_OBJECT) can be split.

Use case: When planning activity quantities (activity RKP2), the composite SAP database field OBJNR in COSS also contains the cost center and activity type. ProCED can provide these two components separately via the constants FD_OBJKEY and FD_OBJACT.

2.5 Posting Relationships from Sender Perspective (source object)

ProCED usually reads data from the receiver's point of view (for example, via field OBJNR in COSS). In cost center planning, however, it can be useful to determine posting relationships from the perspective of the sender (reading activity inputs which have already been planned by the receiver). The goal is to display the "scheduled quantities" to the sender, with the recipient information itemized.

Note:	Reading from the sender's point of view was originally only available via FP Read. However, the ProCED variant offers the advantage that all other planning functions can be mapped very easily via the standard Allevo. For this reason, implementation via ProCED should first be checked in new projects.

In this case you select via the start object (USPOB in COSS)

Characteristics of this function:

- The function is activated via the constant KS_ACTIVITY_MODE; here you have the option of restricting of receiver objects by object type.
- Start object is always a cost center
- The relevant KL objects are determined using COSL, COKL with respect to the version from the start screen of the transaction, and the version for indirect activity allocation for ILV (TKA07-VPDIO)
- Data are read from COSS via the field USPOB (source object)
- There is no aggregation – i.e. all found rows are transferred to the satellite table.
- Sorting of the result list is done via UOBJKEY, UOBJACT, ROBJTYPE, ROBJKEY_EXT, ROBJACT, KSTAR (can partially be changed with the constant SORT_BY_KSTAR).

For this function, the various mapping fields for the source object (see FD_UOBJECT, FD_UOBJTYPE ...) and the recipient object (FD_OBJECT, FD_OBJACT, etc.) are available.

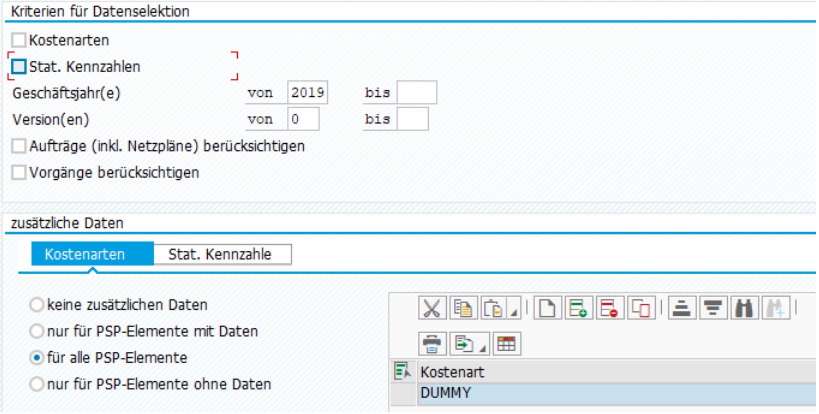
Note:	In this use case, the receiver objects can be different and, in particular, differ from the start object. As of Allevo 3.3, information fields with additional master record information refer to the receiver object. See documentation for the constant KS_ACTIVITY_MODE.

2.6 ProCED for Displaying Project Structure (one row per WBS)

In some cases ProCED is used to read a project structure and to transfer appropriate information per WBS to Excel. In this case real accounting relations based on cost elements or statistical key figures do not play a role (goal is only one row with basic data per WBS).

This list can be generated using any dummy entry for the cost element; at the same time, please ignore cost element and statistical key figure during data selection.

This is what it could look like:



Kriterien für Datenselektion

□ Kostenarten
□ Stat. Kennzahlen

Geschäftsjahr(e) von 2019 bis
Version(en) von 0 bis

□ Aufträge (inkl. Netzpläne) berücksichtigen
□ Vorgänge berücksichtigen

zusätzliche Daten

Kostenarten | Stat. Kennzahle

○ keine zusätzlichen Daten
○ nur für PSP-Elemente mit Daten
● für alle PSP-Elemente
○ nur für PSP-Elemente ohne Daten

Kostenart
DUMMY

Figure 2.2: Selection criteria for displaying the project structure

In the use case here we speak of "project structure", for a list to other object types the procedure is of course equally applicable.

3 Variants for Calling ProCED

Depending on how the ProCED is called, different business scenarios can be mapped. The following sections should provide an overview, but certainly cannot represent all options.

3.1 ProCED with Call via Allevo MultiPage Mode

Allevo ProCED can also be used in conjunction with Allevo MultiPage-mode: This mode generates a separate Excel worksheet for each planning object in the Allevo initial selection.

Note:	The use of the MultiPage-mode via satellite 0 is not useful since this satellite exists only once in a MultiPage file. Satellites between 01 and 99 should be used instead.

For calling the ProCED in MultiPage mode the following settings have to be considered:

- Transfer of suitable content: The Allevo constant MULTI_PROCED (activation with X) is used to activate the call of ProCED per sheet. ProCED is then called up individually for each object / sheet in the Excel file. Therefore the Excel sheet contains the posting relationships to the object valid there.

- Moments: As usual, the satellite data with the relevant posting combinations must be transferred before reference data is read. In the Allevo satellite assistant, read time 1 is relevant for this; as of 4.0, time 2 is preferable in order to avoid multiple calls (the entry in columns ROS of the satellite assistant corresponds to constant READ_ORDER_SAT).

If ProCED is executed in advance by transaction (i.e. not automatically via the Allevo interface), then in the standard case the posting combinations for the representative object (i.e. for the first cost center in a group, for example) are stored in the satellite table. This ensures that all data is found in the MultiObject mode of Allevo. For MultiPage mode, on the other hand, the data must be stored for the current object in order to read it correctly for each sheet in Allevo. To achieve this, the constant MULTIPAGE_MODE must be activated on the ProCED side.

3.2 ProCED with Call MOWO for Variable Sheet Assignment

In MultiPage mode, the main sheet it Allevo is copied according to the number of objects that are to be edited in one call and for which the planning sheet has the same structure (same object on all sheets).

In the MOW variant of the Allevo Master, it is possible to use an individual Excel worksheet for each object in MultiObject mode (see details in the Allevo Excel manual). Control is via an Excel spreadsheet on the "Masterdata" sheet of the Allevo master.

It may be useful to fill this control table dynamically via ProCED (e.g. to provide the appropriate subordinate levels to be managed on their own sheets). It is mandatory to transfer the data via satellite 00, which is transferred from SAP to Excel directly after calling the master: Only with this satellite subsequent Excel functions for generating the relevant sheets are called in time.

Example	Use case ProCED and MOWO (1) Satellite 00 creates a list with the WBS elements that are responsible for determining the pages. This satellite is transferred first when Allevo Planning is called up.

(2) Optionally, a second satellite transfers the posting relations to be entered per page. READ_ORDER_SAT must be set there. In order to dynamically determine the AllocationSets per sheet, it must be ensured that the associated formulas are executed in time (i.e. after event AFT_LEAD_SAT).

In both cases, the data structures are determined via ProCED; with varying degrees of detail of course.

3.3 ProCED and MOD Mode

ProCED offers similar functions as are available in the MOD-mode of Allevo planning: in both cases, posting relationships are provided for planning in the Allevo master via a satellite. However, a combination of MOD and ProCED can also be useful for these satellites.

Examples

(1) MOD-mode offers extended selection options when calling Allevo planning (e.g., via a cost center group). The integration of ProCED can be used to determine details of the relevant posting relationships that are not available in MOD (for example, information regarding the hierarchy in the project).

(2) Planning should be called for cost centers, and via ProCED, posting relationships of assigned orders are determined (so constant READ_ASSIGNED_OR is set there).

With this combination, the automatic calling of ProCED can be set up via Allevo planning (see next section).

ProCED typically uses a representative object for data transfer. Alternatively, the data read by ProCED can also be saved and transferred to the respective object. To do this, the ProCED constant MULTIPAGE_MODE must be set as well as the constant GRP_READ_SATxx in the Allevo layout.

Detailed documentation for setting up the combination of ProCED and MOD-mode can be found in the Allevo Excel manual.

Special case: "Loose" linkage with transfer group to ProCED

The previous description assumed that both the MOD and ProCED provide posting relationships for further processing via the Allevo master. The case "loose" linkage is to state that only ProCED provides the posting relationships via a satellite. The MOD constant is only set to enable the entry of a group (e.g. cost center group), which is then to be transferred to ProCED.

Note:

As of 4.0 the way via MOD is still useful to allow the processing of a group or multiple selection. In this case, the constant FLEX_FIELDS offers the additional option of restricting the input directly to individual fields (e.g. only allowing input for the group).

Associated steps in interaction with MOD:

- The MOD must be active only for the group to appear on the initial screen; however, the associated MOD satellite is not included in Excel.
- By default, the MOD only observes objects with status: this restricted list is then also passed on to ProCED.
- Up to Allevo 3.5 additionally constant PROJ_HIDE_FIELD = 3 must be set, so that the group is also transferred to ProCED (the setting has otherwise no further meaning).

In the standard case, the data is provided via the respective representative object of the group.

Alternative: If a transfer of the posting relationships per object is desired, the constant MULTIPAGE_MODE must be set in ProCED. In this case, GRP_READ_SATxx must also be set for the ProCED satellite in Allevo main planning.

Please note: If READ_ASSIGNED_xx is set at the same time, MULTIPAGE_MODE will not bring the expected result (posting relationships of subordinate objects are always determined in the block and subsequent assignment to the main object is therefore not possible).

3.4 Call ProCED Directly from Allevo Main Planning (workplace)

ProCED can be integrated into the main Allevo program in such a way that data is provided up to date with every call (transaction /ALLEVO/PROCED therefore does not have to be executed manually). In individual cases, e.g. long lists with a lot of data, performance can also become critical with this online procedure.

When called, suitable selection parameters are automatically passed to ProCED; e.g. the relevant project number, in current ProCED versions also extended selection parameters like group or multiple selections are considered. Please note:

- Allevo versions from 4.0 allow very flexible object selection on entry (e.g. via group or multiple selection); alternatively, selection can also be made via tree. To ensure that all relevant objects are also transferred to ProCED, the constant GRP_READ_SATxx must be set. Without this constant, Allevo only passes the representative object to ProCED (this restriction is intended to prevent selections being accidentally triggered over too large a set of objects, similar to the reading of satellite data). See section 3.4.4.

- The calling page decides which objects of the initial selection in ProCED are to be considered. This allows e.g. the consideration of Allevo status information: For a cost center group only those cost centers are requested from ProCED for which a plan status is also stored (exception is of course Allevo constant STATUS_READ_ALL).

- With this call, the list of the found posting relations is also transferred directly without "detour" via the stored satellite (no explicit writing/reading of the database table and thus performance improvement compared to previous program versions).

- If ProCED is called for several objects at the same time and the separation on the Excel side is done via COOBJECT, the constant MULTIPAGE_MODE should also be set on the ProCED side.

- In the standard case, Allevo transfers the information on the planning year and version as stored in the TimeSet of the main planning (e.g. from CX_WW). On this basis, ProCED then selects the relevant posting relationships. Optionally, it may be useful to transfer a list of all years and versions that are stored in Allevo TimeSets for reading data (in order to expand the selection in ProCED as well, see Allevo constant PROCED_PARAM).

The settings required for the direct call on the Allevo or ProCED side are described in the following two sections.

3.4.1 Activation in the Constants of the Allevo Main Program

If the constant ACTIVE_PROCED is set in the Allevo main program, the ProCED program is automatically executed in the background before the planning file is opened.

Note:	The constant ACTIVE_PROCED is activated as follows: An X is entered in "Value from" if the satellite table 0 is used. Other satellite tables are to be activated by "0" and "1" combinations in "Value to". Please use the assistant for constant maintenance (CED column).

In Allevo versions prior to 4.0, the transfer of posting relationships from ProCED to the Allevo master always takes place via a satellite: Even with direct integration, the ProCED posting combinations were first stored in the satellite and then read from there again. In current Allevo versions, direct transfer took place at runtime: In this case, the satellite is only required for defining the transfer structure.

Originally only instead of satellite 0 was intended for the transfer of the data from ProCED: there automatically correct sequence is ensured with transfer of the satellite data to Excel (the transfer of the posting relations must take place before the reading of reference data). If satellites 01 to 99 (instead of satellite 0) are to be used for ProCED, this early reading must be explicitly set via constant READ_ORDER_SAT (see F1 help for details). Again, use the satellite assistant in Allevo constant maintenance (the switch in the ROS column is relevant for the reading order).

ProCED can store data to a representative object (e.g. project representative for all related WBS elements; see section 2.1 below). When you enter Allevo planning via this object, the posting relationships for all associated objects are read automatically.

Alternatively, access to Allevo planning can also take place for several objects simultaneously (e.g. multiple selection or group, possibly also in the tree). In this case, the Allevo constant GRP_READ_SATxx must also be active so that all objects are also taken into account: This applies to reading the data from the satellite but also to direct integration of ProCED.

3.4.2 Activation in the Constants of the ProCED

On the ProCED side, the constant ALLEVO_ACTIVE must also be activated with an X in the field "Value from". For "Value to", the Allevo planning layout for which automatic execution is to take place must be entered. Several entries are possible here (separated by commas).

When entering via multiple selection or Allevo group, the constant MULTIPAGE_MODE should also be active so that the posting relationships are transferred individually for each object (read then via Allevo constant GRP_READ_SATxx, see note above).

3.4.3 Special Features for Object Type PR

When working with projects and WBS elements, it is always a question of how to resolve the associated hierarchy. This is an issue when calling transaction /ALLEVO/PROCED (see section 1.4.1), but also applies when controlling via Allevo main planning. The following cases must be distinguished again (initially without using the tree in Allevo):

- When entering with project, Allevo determines all associated planning-relevant WBS elements and transfers them to ProCED.

- Even when entering with a group or multiple selection, Allevo determines all relevant WBS elements and transfers them as a list.

- When entering with a single WBS element, the constant PSP_MULTI decides whether ProCED should only consider this WBS element or also all subordinate ones. This special function is also applicable when selecting a WBS in the Allevo tree.

In all cases, Allevo checks the status of the respective WBS elements: Possibly the hierarchy is not complete, but adapted to the requirements of the planning.

The list of determined posting combinations is transferred directly via the main memory: In previous versions, ProCED had stored in the deposited satellites and Allevo had read from there again. With this direct transfer, the representative object is also no longer important.

If, alternatively, the data is to be provided in advance via transaction, the constant ACTIVE_PROCED must of course not be set in the Allevo (instead, early reading must be set via constant READ_ORDER_SAT). In this case, enrichment functions could also be used.

3.4.4 Posting Combinations at Tree in the Allevo Panel

When called via the tree in the Allevo panel, the constant GRP_READ_SATxx usually decides what should happen when a group within the hierarchy is selected in the tree.

From Allevo 4.0, this constant also determines which objects are queried via ProCED (possibly again all objects in a group depending on the status). file, selections are also made via dependent object types, as they can be defined via the ProCED constant READ_ASSIGNED_xx.

3.5 Call via Report Transaction /ALLEVO/PROCED_REP

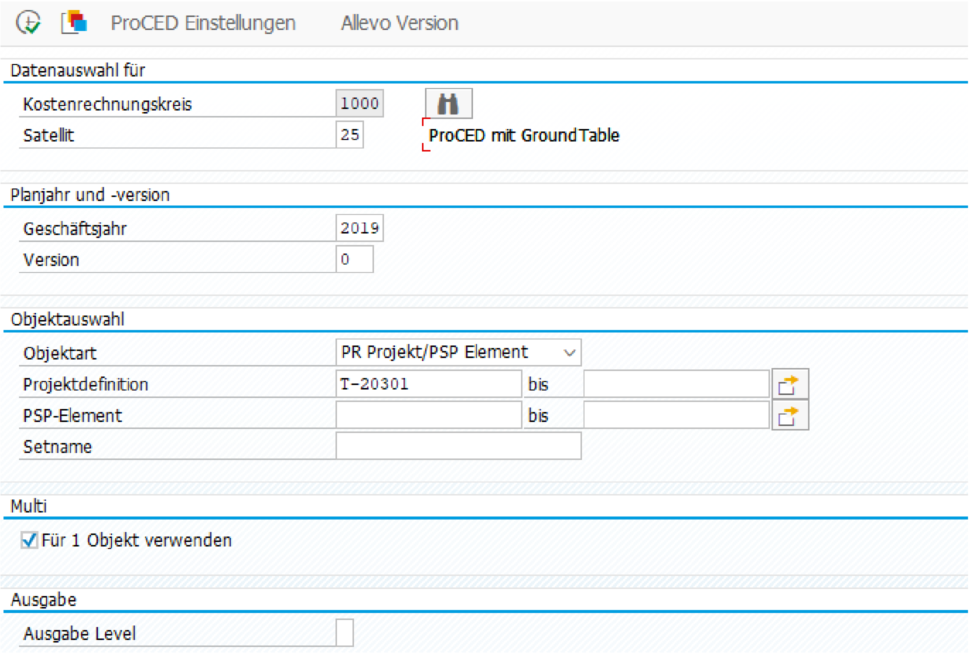
The transaction /ALLEVO/PROCED described so far has been developed dialog-oriented and combines functions that are useful for the setup. Additionally with /ALLEVO/PROCED_REP a transaction is available to let the data provision run in a reporting mode. The report variant has two main advantages:

- Recurring calls/selections can be saved in a report variant.

- Execution in the background is possible (e.g. for nightly updated data provision).

Note:	In previous Allevo versions, there were different transactions for the same call, depending on the object type (e.g. ALLEVO/PROCED_PR). As of 4.0, it is a central transaction; which also starts a common report in the background.

As usual, the selection can again be made via a group and any number of objects (also with multiple selection).



Datenauswahl für

Kostenrechnungskreis 1000
Satellit 25
ProCED mit GroundTable

Planjahr und -version

Geschäftsjahr 2019
Version 0

Objektauswahl

Objektart PR Projekt/PSP Element
Projektdefinition T-20301 bis
PSP-Element bis
Setname

Multi

☑ Für 1 Objekt verwenden

Ausgabe

Ausgabe Level □

Figure 3.1: Call ProCED via reporting transaction

The different selection features can be used simultaneously: e.g. selection of objects (also via multiple selection) and simultaneous entry of a group. The program internally generates a list of all relevant objects and starts the data determination with it. For each object, the program checks whether a 1:N assignment is stored in the ProCED settings; if necessary, all objects of the group stored there are read and the associated data are all saved under the representative object.

Another control is possible via switch "Use for 1 object" for groups: In this case, the representative object must be entered additionally as a single object.

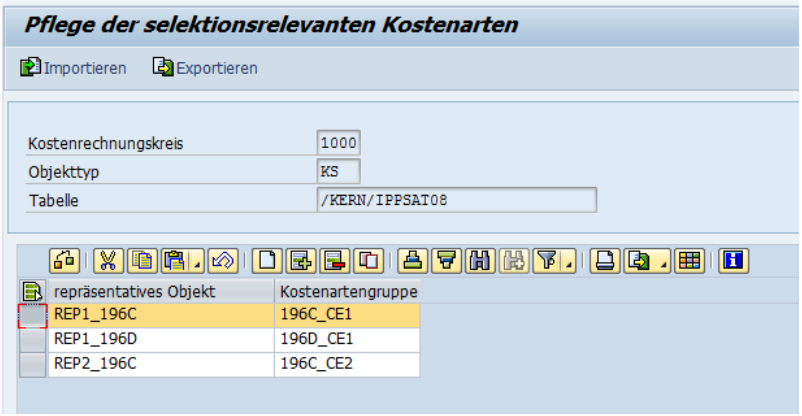
Note:	Special handling of the "Use for 1 object" option for object type PR: For entries at project level, a representative object is determined for each project if "Use for 1 object" is set: the posting relationships found are therefore always saved for the project as a whole. If the option is not set, all projects are completely exploded and the posting relationships found are determined for each WBS element. When accessing via WBS elements, "Use for 1 object" only has a summarizing effect if the WBS elements are in the same project. An entry of a group to projects the option "Use for 1 object" is always ignored (for the other object types objects and groups can be mixed).

When a project number is entered, the posting data is automatically saved to the representative WBS number.

4 Special Conditions of Execution

4.1 Restrictions on Specific Cost Elements or Groups

For the selection of historical postings it can be useful to limit the list of relevant cost elements (or accounts for object type PC) from the beginning. ProCED offers this function via button |Restr. cost element group|.



Pflege der selektionsrelevanten Kostenarten

Kostenrechnungskreis	1000
Objekttyp	KS
Tabelle	/KERN/IPPSAT08

repräsentatives Objekt	Kostenartengruppe
REP1_196C	196C_CE1
REP1_196D	196D_CE1
REP2_196C	196C_CE2

Figure 4.1: Assignment of a cost element group to the representative object

Depending on the current object with which ProCED is started, a suitable cost element group can be stored (or account group when accessing via object type PC): As a result, only the cost elements/accounts of this group are taken into account in the data selection. In this case, the entry object is entered as a representative object, as it is also used in Allevo Planning.

Optionally, a * can be entered in the column for the representative object: Then ProCED works with the assigned cost element group by default (if no explicit assignment exists for the current object).

For other restrictions (especially on objects) see selection criteria in the ProCED initial transaction (see section 0). There you can also deliberately create posting combinations for certain cost elements/accounts, e.g. if there is no historical data yet.

4.2 Avoidance of Unnecessary Runtimes through Period Comparison

Note: this function is **no longer supported by** Allevo 4.0 (data is transferred there directly via ABAP interface without intermediate storage in the satellite).

To avoid unnecessary runtimes, it is possible to specify that ProCED, when it’s called up via the Allevo main program, only redetermines posting relationships if changes are to be expected in the data. This function is activated via an X in the field "Value to" of constant FD_JAHRPER.

Background: Usually the to-period of CX_RR (or the main TimeSet for reading) contains the last closed period. Since the past data do not (should not) change in closed periods, a new selection via ProCED would not provide any new findings.

For this reason, the change check is based on CX_RR (or the main TimeSet for reading). Sequence:

- Via constant FD_JAHRPER ("value from") ProCED remembers the time of the last execution with reselection of posting relations (stored in format MMM.YYYY, i.e. on monthly basis).

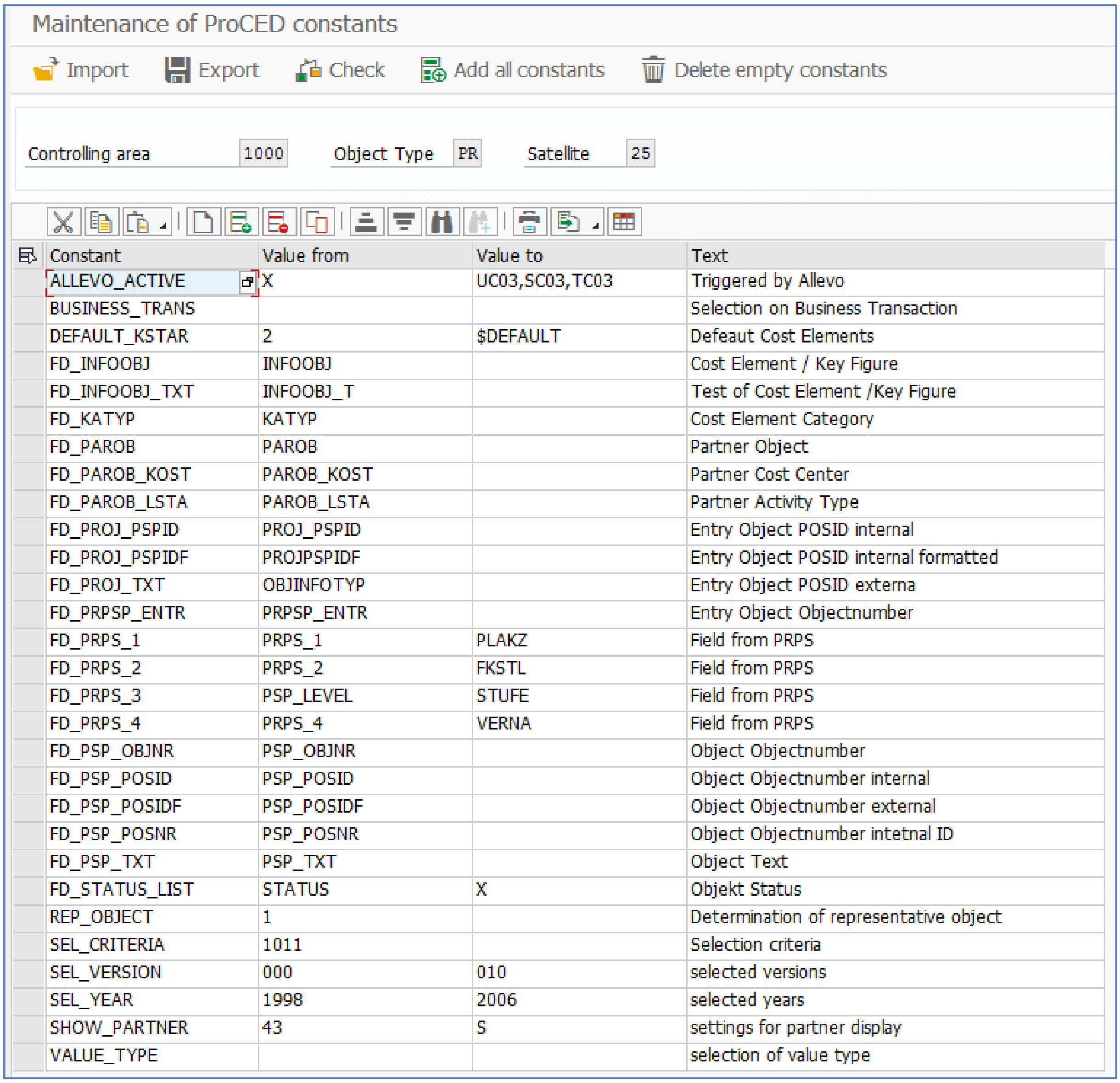
- Each time ProCED is called from the Allevo main schedule, a check is made to see whether the to-period of CX_RR is greater than the period of the last ProCED run.

5 Necessary Basic Settings: The Constants

5.1 Input Constants

Many functions in ProCED are controlled by constants, which are entered depending on the object type and satellite table.

To be able to define the constants for a specific satellite table, in the Allevo constants the satellite must be activated for the respective object type.



Controlling area 1000 Object Type PR Satellite 25

Constant	Value from	Value to	Text
ALLEVO_ACTIVE	X	UC03,SC03,TC03	Triggered by Allevo
BUSINESS_TRANS			Selection on Business Transaction
DEFAULT_KSTAR	2	$DEFAULT	Defeaut Cost Elements
FD_INFOOBJ	INFOOBJ		Cost Element / Key Figure
FD_INFOOBJ_TXT	INFOOBJ_T		Test of Cost Element /Key Figure
FD_KATYP	KATYP		Cost Element Category
FD_PAROB	PAROB		Partner Object
FD_PAROB_KOST	PAROB_KOST		Partner Cost Center
FD_PAROB_LSTA	PAROB_LSTA		Partner Activity Type
FD_PROJ_PSPID	PROJ_PSPID		Entry Object POSID internal
FD_PROJ_PSPIDF	PROJPSPIDF		Entry Object POSID internal formatted
FD_PROJ_TXT	OBJINFOTYP		Entry Object POSID externa
FD_PRPSP_ENTR	PRPSP_ENTR		Entry Object Objectnumber
FD_PRPS_1	PRPS_1	PLAKZ	Field from PRPS
FD_PRPS_2	PRPS_2	FKSTL	Field from PRPS
FD_PRPS_3	PSP_LEVEL	STUFE	Field from PRPS
FD_PRPS_4	PRPS_4	VERNA	Field from PRPS
FD_PSP_OBJNR	PSP_OBJNR		Object Objectnumber
FD_PSP_POSID	PSP_POSID		Object Objectnumber internal
FD_PSP_POSIDF	PSP_POSIDF		Object Objectnumber external
FD_PSP_POSNR	PSP_POSNR		Object Objectnumber intetnal ID
FD_PSP_TXT	PSP_TXT		Object Text
FD_STATUS_LIST	STATUS	X	Objekt Status
REP_OBJECT	1		Determination of representative object
SEL_CRITERIA	1011		Selection criteria
SEL_VERSION	000	010	selected versions
SEL_YEAR	1998	2006	selected years
SHOW_PARTNER	43	S	settings for partner display
VALUE_TYPE			selection of value type

Figure 5.1: Constants, here object type "PR" for satellite table

A list of the available constants can be called via F4 Value help in the input fields of the column "Parameter". In addition, F1 help is also available where the function of all constants is explained in detail. This help is set up via the file management in the Allevo main module (see symb. file name ALLEVO_HELP).

Note:	Constants starting with FD_ always require an assignment of fields of the respective satellite table. Accordingly, F4 in column "Value from" provides a list of the fields that are available in the satellite. These fields therefore pass content to the satellite: If content from additional master data is also required, it may be useful to use the Allevo satellite functions for enrichment (see Allevo SAP manual, earlier keyword TFR Rules). The other constants control the provision of further information of the reading process.

ProCED supports the transfer of setting to other systems via functions for upload and download. The function "Check constants" checks if e.g. the field names are correct.

5.2 General Constants

Parameter	Meaning
ALLEVO_ACTIVE	Setting for automatic execution from the Allevo main program and specification of the relevant planning layout (see section 3.4)
BUSINESS_TRANS	When entering the CO business transactions under "Value from", the selection of the data is restricted to the specified business transactions. Via "Value to" a list of transactions can alternatively be given, which are NOT to be considered.
DEFAULT_KSTAR	In addition to the cost elements of the actual selection result, further cost elements can be taken into account via this parameter. The following options are available: ▪ Value from = 0: no additional cost elements ▪ Value from = 1: The WBS elements for which cost elements have already been determined via the selection are also assigned the cost elements entered under "Value to". ▪ Value from = 2: the cost elements specified in "Value to” are assigned to all WBS elements ▪ Value from = 3: only for those objects for which NO cost elements were determined by selection, the cost elements specified at **Value to** are assigned. The definition of these constants is done in the ProCED start dialog under "additional data" (see section 1.3).
DEFAULT_STAGR	Analogous to DEFAULT_KSTAR, only for statistical key figures
FD_INFOOBJ	Field name for the cost element or statistical key figure
FD_INFOOBJ_TXT	Field name for the text of the cost element or statistical key figure If the associated cost element was added as a default value via the start dialog, the text is only filled if a valid master data record is stored.
FD_INFOOBJ_LTXT	Field name for the long text of the cost element or statistical key figure
FD_JAHRPER	Field name for the annual period of the last update (see section 4.2)
FD_KATYP	Field name for the cost element type or the stat. Key figure type (not for PC).
FD_OBJINFOTYP	Field name for the info type of the object. Possible values: "KSTAR" for cost element, SAKNR account profit center, or "STAGR" for statistical key figure (or DEFAULT in front of it if there are no postings).

Parameter	**Meaning**
FD_PAROB	Field name for the partner object (see section 2.4 on the influence on the selection).
FD_PAROB_COST	Field name for the cost center quota of the partner object
FD_PAROB_LSTA	Field name for the activity type quota of the partner object
FD_SOURCETAB	Field name of the data record source (possible contents: COSS, COSP, COSR and empty for default KSTAR or STAGR).
FD_VRGNG	Business transaction (if set, a separate row with the relevant posting characteristics is provided for each transaction).
SEL_CRITERIA	In the selection screen of the transaction /ALLEVO/PROCED, specific characteristics that are not available when calling directly via Allevo planning can be set in the range "data selection criteria". This refers to the activation button for the cost type, the statistical key figure, as well as the consideration of orders and business transactions. The entry is also taken into account when calling via Allevo planning.
SEL_VERSION	Version with which the data selection is to take place. The entries from the start dialog are automatically taken over and stored here (here version from...to). When called directly from Allevo, the parameter can be taken from the TimeSets relevant there for reading (see Allevo constant PROCED_PARAM).
SEL_YEAR	Specification of the year with which the data selection is to take place. The entries from the start dialog are automatically taken over (data selection, here year from...to). The selection feature can also be derived from Allevo TimeSets, just like SEL_VERSION.
SHOW_PARTNER	With this constant you can define the cost element type for which the partner data is selected. In "Value from", the cost element type is entered (with Profit Center this is the business transaction), in "Value to", the credit/debit indicator is entered. Multiple entries should be separated with a comma.
VALUE_TYPE	With the entry of the value types in "Value from", the data selection is restricted to only these data types (for PC object types this is the record type).
NODE_AGGREGATION	Aggregation to cost element groups
SORT_BY_KSTAR	Sorting the generated list of posting relations
DEBIT_CREDIT_IND	Selection via defined debit/credit indicators
LANGU_KSTAR_FIX	Define the language for the text of cost elements and statistical key figures independent of the logon language. The entry is made as a two-character language key in the field "Value from".
FD_FIX_VALUE_X	In special cases it may be useful to specify the contents of a satellite field (e.g., for supporting Allevo rule planning). When entering the constant, 'X' is entered as a number between 0 and 9 (10 total entries are possible for this constant).
FD_UOBJECT	Field name for the composite abbreviation for the source object
FD_UOBJTYPE	Field name for the object type for the original object
FD_UOBJKEY	Field name for the origin cost center

Parameter	Meaning
FD_UOBJACT	Field name for the activity type in the source object

Depending on the use case, additional information is taken from Allevo constants; e.g.

- MULTIPAGE_MODE as the basis for determining a representative object
- LEDGER for the specification of the relevant ledger (column "Value from").

5.3 Specific Constants for Object Type PR

The following constants are relevant for the determination of postings for WBS elements.

Parameter	Meaning
FD_PRPS_1	Free field selection from the WBS master data (table PRPS). In "Value from" enter the field of the satellite table, in "Value to" enter the field of the PRPS table.
FD_PRPS_2	Free field selection 2 (see FD_PRPS_1)
FD_PRPS_3	Free field selection 3 (see FD_PRPS_1)
FD_PRPS_4	Free field selection 4 (see FD_PRPS_1)
FD_PRPSP_ENTR	Field name for the initial object (project or WBS element). Transfer as internal object number.
FD_PROJ_PSPID	Field name for the external ID of the hierarchically superior project/start object (as in PROJ-PSPID; displayed without formatting rules set in Customizing, e.g. "T3020") Equivalent to FD_PSP_POSID on the WBS level
FD_PSP_OBJNR	Field name for the object number of the (dissolved) WBS element (as in the table field PRPS-OBJNR; usually comprised of "PR" and the internal number of the WBS element in POSNR)
FD_PSP_POSID	Field name for the external ID of the WBS element, e.g., in the table field PRPS-POSID (displayed without formatting rules set in Customizing, e.g. "T3020C01")
FD_PSP_POSIDF	Field name for the external ID of the WBS element, e.g., in PRPS-POSID (displayed without formatting rules set in Customizing, e.g. "T-3020.C01")
FD_PSP_POSNR	Name of the field for the internal ID of the WBS element (e.g. in PRPS-POSNR, numeric with 8 digits)
FD_PSP_TXT	Field name for the text of the WBS element
FD_PSP_LEVEL	Level of the WBS element in the project hierarchy
REP_OBJECT	Settings for the determination of the representative object (see section 2.1.1)
FD_PROJ_TXT	Field name for the short text description of the hierarchically superior project
FD_PROJ_PSPIDF	Field name for the external ID of the hierarchically superior project/start object (as with FD_PROJ_PSPID, but with inclusion of the formatting rules from Customizing, e.g. "T-302.0)
FD_PROJ_1	Free field selection from the project master data (table PROJ). In "Value from" enter the field of the satellite table, in "Value to" enter the field from the PROJ table.

Parameter	Meaning
FD_PROJ_2	Free field selection 2 (see FD_PROJ_1)
FD_STATUS_LIST	Field name in the satellite table, which should contain the system status of a WBS element (from Version 3.3.22 on also the user status). The display is the same as in the respective master data record (e.g. "REL SETC ", language is always EN). This information can be used to filter objects with a certain status on the Excel page. See the detailed F1 documentation for this constant Alternatively, the constant CHECK_STATUS allows you to filter the list of posting elements before it is transferred to Excel.
PSP_ACT_CUMULATE	This constant defines to which level a project should be relevant for Allevo, and if necessary only the WBS elements up to the defined level are written in the satellites. For the determination of the relevant cost elements, the subordinate WBS elements are also considered. To have the correct display in Allevo-MultiObject, the constant of the same name should also be set. Then, the cumulative costs of all of the subordinate levels appear for the WBS of the selected level. Dann erscheinen die Kosten aller untergeordneten Ebenen kumuliert zum PSP der gewählten Stufe. The level must be entered in the column "Value from" (differing from the entry of the constant of the same name in Allevo).
READ_ASSIGNED_OR	Evaluation of the orders assigned to the current WBS element (see section 0): "Value from" activates (with "X") this functionality; "Value to" contains the selection variant in which additional criteria for the selection can be entered, if needed.
FD_IMZO	Investment program position (internal display)
FD_IMZO_EXT	Investment program position (external display)

5.4 Specific Constants for Object Type KS

The following constants are relevant for the determination of postings for cost centers.

Parameter	Meaning
FD_CSKS_1	Free field selection from the cost center master data. In "Value from", enter the field of the satellite table, in "Value to" enter the field from the CSKS table.
FD_CSKS_2	Free field selection 2 (see FD_CSKS_1)
FD_CSKS_3	Free field selection 3 (see FD_CSKS_1)
FD_CSKT_K	Short text of the cost center (language-dependent)
FD_CSKT_L	Long text of the cost center (language-dependent)
FD_KOSTL	Cost center key
READ_ASSIGNED_OR	Evaluation of the orders assigned to the current cost center (see section **Error! Reference source not found.**): "Value from" defines the relevant field of the cost center in the master data record of the order, "Value to" contains the selection variant in which additional selection criteria can be entered.

Parameter	Meaning
READ_ASSIGNED_PR	Same function as with READ_ASSIGNED_OR, but here for access to the assigned WBS elements (see section **Error! Reference source not found.**).
FD_VALID_TO	Date of the highest validity
FD_LOCK	Plan block for cost center
FD_STATUS_LIST	Since the master record for the cost center does not contain status information such as order or WBS element information, information about locks is shown here. For example, PKZKP if there are no locks for "Primary cost plan" (PKZKP is the corresponding field name in table CSKS). Alternatively, the constant CHECK_STATUS allows you to filter the list of posting elements before it is transferred to Excel.
KS_ACTIVITY_MODE	Activates selection of posting relationships from sender view (corresponds to selection via source object in field USPOB of COSS)
FD_OBJECT	Field name for the composite receiver object number
FD_OBJTYPE	Field name for the receiver object type
FD_OBJKEY	Field name for the receiver object in internal representation
FD_OBJKEY_EXT	Field name for the receiver object in external representation
FD_OBJACT	Field name for the activity type at the receiver object, useful especially when evaluating activity relationships, e.g. from the sender view (see section 2.5).

5.5 Specific Constants for Object Type OR

The following constants are relevant when determining postings for internal orders.

Parameter	Meaning
FD_AUFK_1	Free field selection from the order master record (table AUFK). In "Value from" enter the field of the respective satellite table; in "Value to" enter the field from the AUFK table.
FD_AUFK_2	Free field selection 2 (see FD_AUFK_1)
FD_AUFK_3	Free field selection 3 (see FD_AUFK_1)
FD_AUFK_4	Free field selection 4 (see FD_AUFK_1)
FD_AUFK_5	Free field selection 5 (see FD_AUFK_1)
FD_AUFK_K	Short text description for the order (language-dependent)
FD_AUFNR	Order key
FD_STATUS_LIST	Order status with the same display as in the respective master data record (e.g. "REL SETC, language is always EN). Can also be used for filtering objects with a particular status from an Excel page. From version 3.3.22 on, the user status is also shown.

Parameter	Meaning
	Alternatively, the constant CHECK_STATUS allows to filter the list of accounting elements before passing it to Excel.

5.6 Specific Constants for Object Type PC (profit center)

ProCED for Profit Center currently does not work for the new General Ledger.

By default, PC data from the "classic" General Ledger (table GLPCT) are read via ledger "8A". In order to read via other ledgers, the Allevo constant LEDGER can be used (* Layout, column "Value to"): multiple entries are also possible (comma separated).

The following constants are relevant for the determination of postings for Profit Center.

Parameter	Meaning
FD_CEPC_1	Free field selection from the master record for Profit Center (table CEPC). In "Value from" enter the field of the respective satellite table, in "Value to" enter the field from the CSKS table.
FD_CEPC_2	Free field selection 2 (see FD_CEPC _1)
FD_CEPC_3	Free field selection 3 (see FD_CEPC _1)
FD_CEPC_4	Free field selection 4 (see FD_CEPC _1)
FD_CEPC_5	Free field selection 5 (see FD_CEPC _1)
FD_CEPCT_K	Short text description for Profit Center (language-dependent)
FD_CEPCT_L	Long text description for Profit Center (language-dependent)
FD_PRCTR	Profit Center key
READ_ASSIGNED_KS	Evaluation of the cost centers that are assigned to the current Profit Center (see section **Error! Reference source not found.**): "Value from" activates this function (with "X"), "Value to" contains the selection variants where additional selection criteria can be entered.
READ_ASSIGNED_OR	Evaluation of the orders assigned to the current Profit Center (see section **Error! Reference source not found.**): "Value from" activates this function (with "X"), "Value to" contains the selection variants, where additional selection criteria can be entered.
READ_ASSIGNED_PR	Same function as with READ_ASSIGNED_OR, but here for access to the assigned WBS elements (see section **Error! Reference source not found.**).
FD_ELIM	Field name for the elimination Profit-Center (only GLPCT)
FD_ELIM_TXT	Field name for the short text description of the elimination Profit-Center
NEW_GL	For object type PC, reading is done from table GLPCT by default. This constant switches to the new General Ledger; thereby reading is done with the table FAGLFLEXT (enter X in "Value from").
FD_VALID_TO	Date of the highest validity
FD_STATUS_LIST	Since the master record for Profit Center doesn’t contain status information such as orders or WBS elements, only the field "Lock Indicator" is evaluated in the

Parameter	Meaning
	Profit Center master record (CEPC-LOCK_IND), in order to mask locked Profit Centers. Alternatively, the constant CHECK_STATUS allows you to filter the list of posting elements before it is transferred to Excel.

6 Display of Hierarchy Levels for Cost Centers, Orders and Profit Centers

As long as Allevo ProCED is executed via the object types KS, OR, or PC with the specification of a group, the fields in which the respective hierarchy structures of the selected group are entered can be defined in the satellite table. Thus, for each object, you can see which groups it is assigned to.

Parameter	Meaning
FD_HIER_1	Key of the node on the top level
FD_HIERT_1	Text of the node on the top level
...	...
FD_HIER_n	Key of the node on the n-th level
FD_HIERT_n	Text of the node on the n-th level

Note: When ProCED is called via a single object, no information about the hierarchy is available.

7 Defaults for the Field Definitions in the Allevo Satellite

It is recommended to use the following field definitions when defining the appends or structures.

Field content	Corresponding parameter in the constants	Data element (component type)	or possible direct type
Description internal order	FD_AUFK_K	AUFTEXT	CHAR, length 40
Order number	FD_AUFNR	AUFNR	CHAR, length 12
Cost center description	FD_CSKT_K	KTEXT	CHAR, length 40
Cost center long text description	FD_CSKT_L	LTEXT	
Cost element or statistical Key figure	FD_INFOOBJ		CHAR, length 10
Text of the cost element or statistical key figure	FD_INFOOBJ_TXT		CHAR, length 40
Year period of the last update	FD_JAHRPER	JAHRPER	NUMC, length 7
Cost element type or statistical key figure type	FD_KATYP	KATYP	CHAR, length 2
Cost center number	FD_KOSTL	KOSTL	CHAR, length 10
Type of object ("KSTAR" for cost center, "STAGR" for statistical key figure)	FD_OBJINFOTYP		CHAR, length 10
Partner object	FD_PAROB	PAROB	CHAR, length 22
Cost center quota in partner object (only for KST)	FD_PAROB_COST	KOSTL	CHAR, length 10
Activity type quota in partner object (only for KST)	FD_PAROB_LSTA	LSTAR	CHAR, length 6
Start object (if a project was specified in the screen, the project will be shown here, otherwise the WBS element from the field "WBS element")	FD_PRPSP_ENTR	PS_INTNR	NUMC, length 8
Start object (if a project was specified in the screen, the project will be shown here, otherwise the WBS element from the field "WBS element")	FD_PROJ_PSPID	PS_PSPID	CHAR, length 24
WBS element of the project hierarchy, object number	FD_PSP_OBJNR	J_OBJNR	CHAR, length 22
WBS element of the project hierarchy (external ID without formatting)	FD_PSP_POSID	PS_POSID	CHAR, length 24

Field content	Corresponding parameter in the constants	Data element (component type)	or possible direct type
WBS element of the project hierarchy (external ID with formatting)	FD_PSP_POSIDF	PS_POSID	CHAR, length 24
WBS element number in the project hierarchy (internal counter)	FD_PSP_POSNR	PS_POSNR	NUMC, length 8
WBS element of the project hierarchy, short text description	FD_PSP_TXT	PS_POST1	CHAR, length 40
Level of the WBS element in the project hierarchy	FD_PSP_LEVEL	PS_STAGE	INT1, length 3
Display of the table from which the data records were read (e.g. COSR for statistical key figures, COSP/ COSS for cost elements; or GLPCT/FAGLFLEXT for PC).	FD_SOURCETAB	TABNAME	CHAR, length 30
Hierarchy nodes	FD_HERE_...	GRPNAME	CHAR, length 15
Hierarchy nodes, text	FD_HIERT_...	BAPISET_DESCRIPT	CHAR, length 40
Start object (like FD_PROJ_PSPID, but with additional expense formatting according to the customizing)	FD_PROJ_PSPIDF	PS_PSPID	CHAR, length 24
Status info (OR and PR)	FD_STATUS_LIST	CHAR15	Text field