



Flexible Planning

ALLEVO

User manual

Valid from Allevo 4.2

Flexible Planning for Activities, Statistical Key Figures, Budgets, Primary and Secondary Costs

The Allevo "Flexible Planning" (FP) module is used to record plan data for concurrent planning objects (co-/sub-postings).

Typical example: A cost element manager plans insurance costs per cost center across the entire company. To do this, he enters the relevant cost center-cost element constellations in a satellite table. Allevo's own follow-up process "Flexible Planning" identifies the combinations to be posted in the satellite, the business transaction to be executed and the amounts to be posted (quantities, values, prices; as annual or monthly data; differentiated by plan versions, months as well as years, etc.) and makes the posting via BAPI.

Content

1	Allevo Flexible Planning (Overview)	4
2	Program Start and Authorizations	6
3	Constants and Program Settings.....	8
4	The Planning Mode	33
5	The Reading Mode (FP Read)	38
6	Suggestions for Field Definitions in the Satellite.....	43



Table of Contents

1	Allevo Flexible Planning (Overview)	4
1.1	Data in an Allevo Satellite	4
1.2	Application of Flexible Planning	4
1.3	Interaction with Allevo COPA Interface	5
2	Program Start and Authorizations	6
2.1	Starting Flexible Planning	6
2.2	Execution Scheme	6
2.3	Constants	7
2.4	Cancellation of Data Records	7
2.5	Authorizations	8
3	Constants and Program Settings	9
3.1	Definition of Constants	9
3.2	Determination Rules (e.g. for the planning year)	10
3.3	General Constants	11
3.4	Field Mapping per Posting Procedure	14
3.4.1	Activities, Capacities and Activity Prices (AT)	14
3.4.2	Primary Costs (PC)	15
3.4.3	Billing Internal (OP)	17
3.4.4	Direct Activity Allocation (DA)	17
3.4.5	Secondary Order Settlement (OS)	18
3.4.6	Statistical Key Figures (SK, KF)	19
3.4.7	Budget and Total Cost Planning (BP, BD)	19
3.4.8	Profit Center Planning	24
3.4.9	Balance Sheet Planning via FP	25
3.4.10	Allevo Objects (object type KX)	26
3.5	Other Program Settings	26
3.5.1	Multi-Year Planning (subsequent years in one row)	26
3.5.2	Planning with Different Cost Elements in one Row	28
3.5.3	Period Planning	28
3.5.4	Posting of Individual Monthly Values or Ranges (from/to)	29
3.5.5	Special Case: Periods in Rows	29
3.5.6	Special Periods (13-16)	29



3.5.7	Currencies	29
3.5.8	Delta Planning	30
3.5.9	Sender Selection for Activity Allocation and Sec. Order Settlement	30
3.6	Cancellation Feature for Changed or Deleted Rows	31
3.6.1	Background and Properties	31
3.6.2	Cancellation Procedure	31
3.6.3	Cancellation Table	32
4	The Planning Mode	33
4.1	Execution Options	33
4.2	Running Multiple Schemes	35
4.3	Processing with Automatic Aggregation	35
4.4	Process Data from Multiple Satellites Together (SATELLITE_INCL)	35
4.5	Integration into Allevo Planning Transactions	36
4.6	Background Processing for Performance Improvement	36
4.7	Additional Authorizations / RFC Call	37
4.8	Allevo Status Management	37
4.9	Checks and Error Messages	38
5	The Reading Mode (FP Read)	39
5.1	Overview and Initial Screen	39
5.2	Call from Allevo Workplace	40
5.3	Reading Activity Inputs	40
5.4	Reading Primary Costs	41
5.5	Reading With Individual Value Types	42
5.6	Reading the Data on a Monthly or Annual Basis	42
5.7	Reading Data in Subsequent Years	42
5.8	Relevant Constants	42
6	Suggestions for Field Definitions in the Satellite	43



1 Allevo Flexible Planning (Overview)

1.1 Data in an Allevo Satellite

Satellite tables in Allevo offer much flexibility for merging detailed data from decentralized planning processes, such as cost center planning, into a central data store on the SAP side. There is correspondingly a wide range of use cases. Satellites are for example regularly used to derive and store plan data on activity quantities or primary and secondary costs from a list of detailed measures. Examples:

- The activity quantities of a sending cost center are determined from a list of all receivers and the respective quantities used.
- Depreciation costs are calculated on the basis of the individual investment activities of a cost center.
- The expected costs from projects are derived from a list of planned projects.

In all these examples, the plan data are determined not only for the cost center to be planned, but implicitly also for its partners. In the first example, this is the receiver cost center, in the second the investment order, and in the third the WBS element. Ideally, these objects should automatically be supplied with data. This is exactly what the Allevo Flexible Planning module is applied for.

A satellite is activated in the system via the Allevo Satellite Cockpit: It consists of a two-digit number (so up to 99 satellites can be addressed), a matching description and other characteristics to be entered in the Cockpit. Behind each satellite there is a matching database table: Its basic structure is delivered in the Allevo package. Usually, customer-specific append fields are added to this table in order to be able to store the desired data. Optionally, the entire table can also be provided by Kern AG (GroundTable). For details see "Allevo SAP manual".

1.2 Application of Flexible Planning

Flexible Planning features a Planning Mode and a Reading Mode (see section 0).

The **Planning Mode** applies for the following postings (allocation type in brackets):

- Activities of a cost center with capacity and/or activity prices (AT)
- Primary costs (PC)
- Activity inputs from cost centers, orders and WBS elements (DA)
- Indirect activity inputs through order settlement costs of cost centers with specification of settlement cost element, sending cost center and sender activity type (OS).
- Statistical key figures for cost centers, orders or WBS elements (KF)
- Original budget and cost planning for orders and WBS elements (BP)
- Profit center planning for "old" or new general ledger

Planning can be on a monthly or annual basis, also for multiple years. If cost centers are the receivers of primary and secondary costs or statistical key figures, the posting can be activity-type dependent.



The **Reading Mode** allows you to read already planned

- activity inputs and
- indirect activity inputs through order settlement costs

by specifying the respective receivers (partner objects and/or objects of origin), to transfer them to the satellite and make them available to the sender. This function is only applicable for the main planning year.

Current program versions contain further options, e.g. for the provision of primary costs as a basis for subsequent postings in balance sheet planning.

1.3 Interaction with Allevo COPA Interface

There are various use cases in which SAP COPA planning data is to be posted in parallel at cost element level. There are different settings that support this cooperation.

Background:

- In the standard display for COPA data, value/quantity fields are in columns, the corresponding periods in rows (if period display is required).
- In FP, the opposite is often the case: A posting row contains data on cost element / account, in period display with a corresponding large amount of columns next to each other in the satellite.

Functions to support this type of co-posting:

- The reading function of the Allevo COPA Interface offers a so-called "FP mode" to convert the data from columns to rows. This enables a direct update if the data is enriched via additional functions (e.g. corresponding cost element depending on the value field and further characteristics such as receiver object).

Alternatively, FP can process the representation of several values in one row (with the same information about year, version, period...); settings in FP can help to find the necessary information about the cost element.



2 Program Start and Authorizations

2.1 Starting Flexible Planning

Flexible Planning is an independent ABAP program in the Kern's own naming environment: To open the module in the **Planning Mode**, use the transaction /ALLEVO/FP, to open it in the **Reading Mode**, use /ALLEVO/FPR.

Allevo: Flexible Planning (FP)

Schemes Constants Cancelation list Allevo Version

Process data from

Controlling Area 1000 Invest

Satellite 01 Test PSP

Scheme PSTEST

Select year and version

Fiscal Year 2018

Version 0

Select objects

Object Type KS Cost Center

Cost Center to

Set Name

Select objects as

Sender (Main Object COBJECT)

Receiver (Using FD_RECEIVER)

Show messages

Message Level

Figure 2.1: Start screen of Flexible Planning; here in Planning Mode

The dialogue covers two functions:

- Execution of planning functions depending on the chosen selection parameters such as satellite, scheme, year, version (see section 4).
- Branching to the program settings (via icons in the toolbar), which will be explained in the following sections.

As in the main Allevo program constants control the processing of the data in a satellite. The constants are stored in a so-called "execution scheme" and must therefore be defined before starting the program (see detailed description in section 0). Such a scheme is always dependent on the controlling area and satellite.

Flexible Planning can execute a single scheme within a program call or several in succession with specified sequence (see section 4.2).

2.2 Execution Scheme

The schemes allow setting up more than one valid combination of constants for a satellite and consequently processing data contained in the table according to different criteria.

The creation and maintenance of the execution scheme is performed using the button of the same name on the initial screen of the FP start transaction. A scheme is defined by the following entries:

- Scheme (key field, 6-digit, alphanumeric)



- Order of execution (2-digit, numeric)
- Description (40 characters, text)

The position allows to control the sequence during program execution, in the case that several schemes are simultaneously activated for processing (see notes in section 4.2).

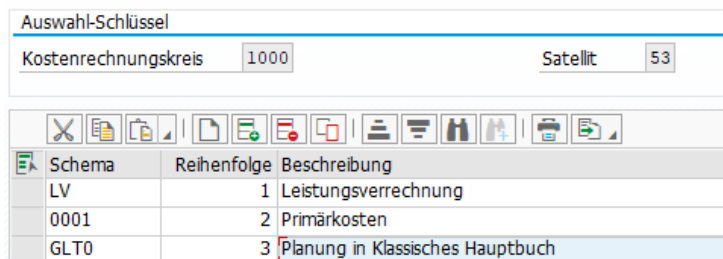


Figure 2.2: Master data for the execution scheme

The position and description of the scheme can be changed later.

Another naming convention is important: For a Read scheme (execution via /ALLEVO/FPR) there must be a "READ" in the abbreviation, other schemes will be ignored (for details on the Reading-Mode see chapter 0).

All other schemes are intended for planning (for details on the Planning Mode see chapter 4).

Note:	A scheme can only be deleted after previously deleting all constants belonging to it.
	Within one call, several schemes can be executed with separate transfer of plan data to SAP (see section 4.2).

2.3 Constants

Suitable constants must be entered for the scheme, with which the actual planning or reading functions are described when the program is executed. These constants particularly link the satellite fields with the processing part, which ensures the transfer to SAP and a suitable posting: It is therefore a matter of specifying in which fields the relevant information is stored.

There can be different constant definitions for each scheme involved; for example, in scheme 1 another table field could be defined as receiver than in scheme 2.

The display of the constants is accessible by pressing the button |Constants| in the initial screen of the FP transactions: Beforehand, the controlling area, satellite and the scheme must be. For further details see the following chapter 0.

2.4 Cancellation of Data Records

Allevo FP's feature of accessing data records in the satellite offers a high degree of flexibility, but also certain risks: What happens e.g. to data records in the satellite that are completely deleted or whose key data is changed if they were previously posted in SAP?

The answer is obvious and clear: The user expects a complete correction, also on the SAP side. So not only the new data record has to be posted, but also the posting to the old record has to be corrected, if necessary.



For these cases, FP offers a cancellation feature that is automatically run through before new data records are transferred to SAP. Currently relevant cancellation data records are displayed via button |Cancellation list|; for further details see section 3.6.

2.5 Authorizations

In general, Flexible Planning considers authorizations stored in SAP for the execution of planning functions. Furthermore, an authorization to execute the respective transaction must be available (e.g. for /ALLEVO/FP).

Customizing functions (change scheme and constants) as well as the maintenance of the cancellation table is usually reserved for employees with administration rights. Correspondingly, individual authorizations regarding the authorization groups ZIPP_FP1, ZIPP_FP2 and ZIPP_FP3 are stored here. Users of the higher authorization groups ZIPP_ADM have access as well (see also notes about the authorization object ZIPP_PROG in the “Allevo SAP manual”).



3 Constants and Program Settings

3.1 Definition of Constants

Constants are particularly used to link satellite fields to the processing program: It is therefore necessary to define the fields containing the relevant information.

The view of constants is accessed via the button |Constants| in the initial screen of FP Transactions: After having specified the controlling area, the satellite and the execution scheme (see Figure 2.1) the following input template will appear:

Parameter in Allevo FP	Wert von	Wert bis	Text
CANCEL_ACTIVE	X		Flag for activation of cancelation. Please use caref...
CURRENCY_TYPE			Währungstyp
FD_ALLOC_TYPE	ZZALLOCTYPE1		Feldname: Verrechnungsart
FD_KSTAR	ZZKSTAR		Feldname: Kostenart /Konto
FD_OBJECT_TYPE	ZZOBJECTTYPE		Feldname: Objektart
FD_QNTY_MNTH_FIX	ZZMONTH		Feldname: Gesamtmenge fix
FD_QNTY_MNTH_VAR	ZZMONTHV		Feldname (ohne Perioden-Suffix): Menge einer Pe...
FD_RCV_ACT_TYPE	ZZFDRCVACTTYPE		Feldname: Empfänger-Leistungsart
FD_RECEIVER	ZZRECEIVER		Feldname: Empfänger
MONTH_ACTIVE	X		Periodengenaue Planung

Figure 3.1: Constants for the execution scheme

The following types of constants are being distinguished here:

- Constants beginning with “FD_“ are linked to a satellite field (field mapping).
- All other constants control the workflow of planning or contain additional information.

A list containing all constants available in the system can be called in the first entry column via F4. After selecting a constant, F1 help is also available, providing additional information on the meaning of the constant; especially the parameters to be set in the columns "Value from" and "Value to". The following applies:

1. Constants with field mapping (i.e. starting with "FD_") expect the name of a field from the selected satellite (usually a field from the append) in column “Value from”. The F4 value help in column "Value from" shows a list of all fields in this satellite.
2. Both “Value from“ and “Value to“ can be individually assigned with other constants. Contents are provided by F1 help.

Note: The F1 constant help is a central Allevo function. The installation is described in the “Allevo SAP manual”.



For some parameters, such as the main planning year, different constants are available for their determination: The content can be permanently assigned, read from the satellite via field mapping or determined via an Allevo TimeSet. The necessary determination rules are described in the next section.

Texts describing the constant are stored in the system but can be modified by the user and saved language-dependently (specifically for the selected scheme).

Up-/download functions are available to transfer constants to a different SAP system (copies all entries regarding a scheme).

Note: Alternatively, it's possible to use the standard ALV functions "Copy text" and "Enter in new row".

The function "check" constants controls for example the validity of entered constant names and satellite fields. In each scheme at least the constants FD_ALLOC_TYPE and/or ALLOC_TYPE, FD_OBJECT_TYPE and/or OBJECT_TYPE, FD_RECEIVER and a FD_QNTY... field must be set up.

3.2 Determination Rules (e.g. for the planning year)

Individual parameters required for the execution of Flexible Planning can be defined in different ways; the main planning year will serve as example here:

- The constant PLANYEAR determines the planning year centrally (entry in column "Value from").
- The planning year can optionally be adopted from a satellite field. The corresponding mapping is performed via the constant FD_PLANYEAR.
- An Allevo TimeSet can further serve as alternative source; this is set via the constant COLDEF.
- The TimeSet can also originate from a satellite field; mapping is performed via the constant FD_COLDEF.

This variety of options for determining a parameter, such as the planning year, requires a rule for the determination: Allevo uses the value of the highest available level of detail.

Note: In the example of the planning year the following sequence applies:
 (1) FD_PLANYEAR, (2) PLANYEAR, (3) via FD_COLDEF, (4) via COLDEF.
 The first constant found in this sequence defines the main planning year. If all constants are missing, the entries according to the selection screen are used as a substitute (when called up in dialog).

Reading the year via a mapping from the satellite offers several advantages, such as:

- Postings can be performed over multiple years without using for separate schemes.
- In the standard case, processing data in the Excel master is also simplified; e.g. only one column needs to be addressed with Allocation Type, which is then relevant for all years.
- If the planning year is drawn via TimeSets, the change to another year is also simplified. Changes then only have to be made centrally in the Allevo TimeSets and Flexible Planning adapts automatically.



Note: The described determination logics are realized for the following characteristics of an Allevo TimeSet: planning year, plan version, period from, period to, delta planning and currency type.

If Flexible Planning is executed in the dialog (i.e. not in the background of Allevo), the appropriate layout must also be known when searching via a TimeSet. In this case, Flexible Planning reads the "Value_to" in the constant ALLEVO_ACTIVE and possibly uses the first entry here (otherwise the * layout for the current object type is used).

3.3 General Constants

Flexible Planning provides a number of constants that can be used regardless of individual posting procedures. The following table gives a short overview. The respective F1 help contains detailed information.

Note: Some constants can be both stored in the scheme and provided via a field mapping. Usually, both constants carry the same name; the "FD_" part of the name is written in brackets in the following table (and the tables below).

The entry (FD_)ALLOC_TYPE for example means:

- that there is a constant ALLOC_TYPE to store the allocation type in the scheme, and
- that there is also the constant FD_ALLOC_TYPE to adopt the allocation type from the satellite via field mapping.

The constants (FD_)ALLOC_TYPE (= allocation type), (FD_)OBJECT_TYPE (= object type) and FD_RECEIVER (= receiver object) are mandatory fields required in every scheme.

Constant	Meaning
ALLEVO_ACTIVE	Activates Flexible Planning for automatic call-up from the Allevo main module by specifying the relevant Allevo layouts.
(FD_)ALLOC_TYPE	<p>Allocation Type (mandatory field):</p> <p>The entry determines which posting function shall be executed in the Planning Mode. Usually, a two-digit abbreviation is used (e.g. DA = Direct Allocation). In the current Allevo version, a number can be used as an alternative, which can facilitate automatic assignment via Excel formulas (e.g. 2 with the same meaning as DA, see detailed list in F1 documentation for constants).</p> <p>If satellite data is to be processed using different schemes (e.g. for different allocation types), then different columns must be used in the satellite for each allocation type (with corresponding mapping for each scheme).</p> <p>When selecting data, satellite rows are ignored if the field addressed via FD_ALLOC_TYPE is empty or an allocation type of "NN" is entered (from Allevo 4.0, rows where a 0 is entered are also ignored, as this is easier to fill as a result of Excel formulas).</p> <p>Special function for satellite data selection: If the constants FD_ALLOC_TYPE and ALLOC_TYPE are entered in the mapping at the same time, then only those data records are processed in the satellite whose allocation type in field FD_ALLOC_TYPE matches the entry in ALLOC_TYPE.</p>



Constant	Meaning
(FD_)OBJECT_TYPE	Object type (mandatory field) When selecting data, satellite rows are ignored if the field addressed via FD_OBJECT_TYPE is empty or an object type "NN" is entered.
FD_RECEIVER	Receiver object (mandatory field): The corresponding satellite field must contain the object for which costs are posted (e.g. the ID of a cost center). Regarding its content, the receiver is equivalent to the start object of a standard Allevo planning transaction. By contrast, the sender object is usually found in the field COBJECT of the satellite (exemptions are described below). In the dialog mode of FP, the field in FD_RECEIVER can also serve as a feature of the selection of satellite data (see notes in chapter 4).
FD_SENDER	Sender object: Normally, the planer of a cost center records own activities, which are to be performed for other objects (e.g. other cost centers) and thus cause costs at the receiver (see FD_RECEIVER). The sending cost center is therefore usually in the field COBJECT of the satellite. Activities of a different cost center can optionally be planned as sender via the starting cost center; this requires an assignment to a different satellite field (see for example section 3.5.9). In most cases, data selection in the satellite is performed via FD_SENDER (characteristic "Initial object/transmitter" in the start screen of the dialog transaction).
(FD_)KSTAR	Specification of cost element / account for value/quantity fields of the main planning year. The stored cost element is also used for subsequent years if no other cost element is stored in the mapping of the corresponding value/quantity fields (FD_VALUE_FIXy, FD_QNTY_FIXy).
CURRENCY_TYPE FD_CURRENCY	Currency of planning: controlling area currency, object currency or transaction currency (see also section 3.5.7)
NO_ZERO_DELTA	Determines that only changed plan data shall be saved.
(FD_)COLDEF	Reference to an Allevo TimeSet to determine one of the following parameters: planning year, plan version, period from, period to, delta planning and currency type (see also 3.2).
(FD_)PLANKOKRS	The plan controlling area is by default specified via the Flexible Planning selection screen (or as a parameter when called from Allevo). In exceptional cases, individual control via constant can be useful: e.g. FD_PLANKOKRS allows postings that are cross controlling area (see F1 help for the constant and section 4.7 on execution via RFC user, if additional authorizations are required).
(FD_)PLANYEAR	Main planning year (can alternatively be determined via Allevo-TimeSet)
(FD_)PLANVERSN	Plan version (can also be determined via Allevo-TimeSet)
PLANPERIOD	Period frame for restricting plan values to individual periods or period ranges, e.g.



Constant	Meaning
FD_PLANPERIODFROM FD_PLANPERIODTO	<p>for posting to a single month or a quarterly value. Plan values outside the period frame keep their old value.</p> <p>The parameters for these constants can also be stored in the Allevo TimeSet.</p> <p>Current Allevo versions also allow special periods (e.g. period 13) if the SAP customizing is set up for this.</p> <p>As an alternative, FP offers the blockwise processing of period values via the so-called "period planning" (constant MONTH_ACTIVE, section 3.5.3).</p>
FD_QNTY_FIXY FD_QNTY_VARY FD_VALUE_FIXY FD_VALUE_VARY COLDEFY YEARY VERSIONY	<p>Parameters for multi-year planning and for planning via multiple cost elements (with entry in one row). See section 3.5.</p>
(FD_)DIST_KEY_FIX (FD_)DIST_KEY_VAR	<p>Keys for the distribution of annual values to periods; constants are usefully applicable only in annual planning. An entry can be made as a constant or via a satellite field.</p> <p>If no distribution key is set via the constants, the posting function searches for a suitable entry for the KL object (table COKL); if no entry is available, the default key "2" will be used for an equal distribution, as in direct planning via Allevo.</p> <p>Available values are provided e.g. in the F4 value list when planning via KP26, or in the detailed F1 documentation.</p>
SIGN_MINUS	<p>The sign of a quantity can be reversed prior to planning by „X“ for SIGN_MINUS. This is useful when planning the receiver debit from the sender perspective; as credit there with inverse sign.</p>
FD_SIGN	<p>As an alternative to SIGN_MINUS, the sign can be read via a satellite field: if the first sign in the belonging satellite field is a "-" all quantities/values are multiplied by -1.</p>
(FD_)FACTOR	<p>A factor with up to 3 decimal places can be entered here. The factor will be included for all quantity/value fields.</p>
DUPLICATE_INCL	<p>Background: If there are several rows with the same characteristics in the satellite (e.g. same cost center, same cost element, etc.) and those are transferred to SAP in succession, then by default only the last posted value appears. However, a sum over all relevant rows is to be expected.</p> <p>With the constant DUPLICATE_INCL active, FP searches the satellite for data records with identical sender/receiver characteristics (duplicates), sums the values and transfers the total to SAP.</p> <p>The constant should only be active when satellite data is selected via the "Sender (main object COBJECT)" option, or when it is called up directly from Allevo planning. It was initially used in activity planning, but applies to all object types.</p>
SAVE_LOG	<p>Allows to write (error) messages into the SAP Application Log. Entries are</p>



Constant	Meaning
	<p>displayed via the SAP transaction SLG1 with selection parameters /KERN/ALLEVO as object and Flexible Planning as sub-object.</p> <p>The log function must be activated explicitly via the constant. Upon calling from Allevo, controlling is performed via the Allevo constant MESSAGE_LEVEL.</p> <p>Special function: Via column "Value to" this constant can also control how Allevo should behave in subsequent steps if several schemes are executed in succession and possibly errors occur in one step.</p>

3.4 Field Mapping per Posting Procedure

The following sections explain, which contents (columns or fields) a satellite must contain in order to be able to serve the corresponding planning fields. These fields must then be assigned to the respective constants.

The columns "Cost center", "Order" and "WBS element" specify the object types that can be addressed to receive the posting via Flexible Planning. Rows contain contents that must be included in the corresponding satellite fields. Please note:

- The entries in quotation marks ("") indicate that exactly this entry is expected for the respective field (column) in the satellite.
- Other entries, such as cost center, refer to the entry of the corresponding SAP cost center etc. in the satellite.
- The program uses the value "2" as the default distribution key.

A detailed description of each constant and its setting parameters is included in the F1 help.

Note:	The following sections describe the used constants depending on the respective posting process. The meaning of a constant can depend on the selected receiver object and therefore on the object type such as KS, OR and PR. Currently, only these object types are explicitly mentioned. From Flexible Planning Version 3.1, posting for business processes (object type BP = Business Process) as receiver is possible (except for budget and total cost planning).
--------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3.4.1 Activities, Capacities and Activity Prices (AT)

Activities of a cost center with capacity and/or activity prices

Constants	Cost center
(FD_)ALLOC_TYPE	"AT" (or "5")
(FD_)OBJECT_TYPE	Object type "KS" or "KL" Please note: When selecting via receiver object (FD_RECEIVER), the correct object type may also have to be entered in the selection screen, i.e. "KL".
FD_RECEIVER	Cost center
(FD_)RCV_ACT_TYPE	Activity type



Constants	Cost center
FD_QNTY_FIX	Activity quantity The sign can be rotated via SIGN_MINUS.
(FD_)KSTAR	Individual allocation cost type of the posting Background: Normally the booked cost element is automatically drawn over the activity type. In current FP versions an individual cost element can also be specified (i.e. similar behavior as in KP26). From 4.0 on, the cost element is also taken into account in period planning.
(FD_)DIST_KEY_FIX	Distribution key: activity (with regard to FD_QNTY_FIX) (must however be included also if only activity prices are being planned)
FD_QNTY_VAR	Capacity
(FD_)DIST_KEY_VAR	Distribution key: capacity (with regard to FD_QNTY_FIX)
FD_PRICE_FIX	Activity price fixed
(FD_)DK_PRICE_FIX	Distribution key: activity price fixed (with regard to FD_PRICE_FIX)
FD_PRICE_VAR	Activity price variable
(FD_)DK_PRICE_VAR	Distribution key: activity price variable (related to FD_PRICE_VAR) If there is no entry in SAP for the distribution key or the existing entry for the activity type is "0", the distribution key "2" is posted (the BAPI does not allow the distribution key "0").
FD_PRICE_UNIT	Price unit (valuation: 1, 10, 100, 1000, 10000).
(FD_)SWITCH_LAYOUT	Switching scheme cost component splitting
(FD_)PLANYEAR	Main planning year
(FD_)PLANVERSN	Plan version
(FD_)EQUI_NO	Equivalence number

Special Notes:

- In the case of multi-year planning, activity prices are planned for the first year only.
- It is not necessary to include all components for activity planning (quantity, capacity, activity prices, distribution keys etc.) in the satellite. If one of these figures is missing, the entry available in SAP will be used.
- Important for cancellation: To reset quantities, they must also be stored in the mapping.

3.4.2 Primary Costs (PC)

This feature is available only for planning on an annual basis. It supports the allocation type PC (primary costs).

Constants	Cost center	Order	WBS element
(FD_)ALLOC_TYPE	"PC" or "1"		
(FD_)OBJECT_TYPE	"KS"	"OR"	"PR"



Constants	Cost center	Order	WBS element
FD_RECEIVER	ID cost center	Order	WBS element
(FD_)RCV_ACT_TYPE	Activity type for activity type-dependent planning		
(FD_)RCV_CCODE	Company code for activity type-dependent planning		
(FD_)RCV_RESOURCE	<p>Indication of the resource when planning quantities.</p> <p>The corresponding cost element must be entered (activity type is only relevant on the receiver side). The price for the calculation of planned costs is found via the resource (see also transaction KPR2 = resource master record and KPR6 = price determination per resource).</p> <p>Allevo can list current resource prices via transaction /ALLEVO/RES_PRICE and possibly provide them in the planning interface via satellite (see notes in the "Allevo SAP manual").</p>		
(FD_)KSTAR	Cost element (either read from satellite or fixed in the scheme)		
FD_VALUE_FIX	Amount (fixed)		
FD_VALUE_FIXy	Amount in multi-year planning or when planning via several cost elements within one row (see notes below or F1 help for the constant).		
(FD_)DIST_KEY_FIX	Distribution key (fixed)		
FD_VALUE_VAR	Variable amount, only for activity-type dependent planning		
(FD_)DIST_KEY_VAR	Variable distribution key, only for activity-type dependent planning		
(FP_)ACTDEP	Activates activity type dependent planning via rules (entry 3 or 4 are possible); see details below.		
(FD_)PLANYEAR	Main planning year		
(FD_)PLANVERSN	Plan version		
(FD_)PLANKOKRS	Plan controlling area		

Special Notes:

- For posting primary costs within a multi-year plan (see constant FD_VALUE_FIXy), in the simplest case the cost element that is stored in the scheme or data record of the satellite is used (see constant KSTAR or FD_KSTAR).
- Alternatively, a posting can be performed using different cost elements if the corresponding cost element is entered in column "Value to" for parameter FD_VALUE_FIXy (y from 0 to 9).
For example, this feature can be applied if data from Allevo COPA planning is to be posted in parallel to suitable cost elements (see also section 3.5.2 on planning via several cost elements).
- The parameters FD_VALUE_FIXy and YEARy (y from 0 to 9) always form a unit which must be stored in the constants.
- Further special features of primary cost posting in multi-year planning are described in section 3.5.1.
- Constant SIGN_MINUS can be used.



Activity type-dependent planning via rules

If planning shall be activity type-dependent, FP can make use of Allevo's regular planning (see details in the Allevo SAP and Allevo Excel manuals). The call via FP is available from version 3.5.25.

Control of the regular planning in FP takes place via the constant (FP_)ACTDEP, in which the type of activity-type dependent planning must be stored: Either 3 or 4 (equivalent to the entries for Allevo column key CY_ADP/CM_ADP). In the background, FP uses the same functions as Allevo standard planning: activated via Allevo constant LSTAR_VARIATOR (see also F1 documentation there).

3.4.3 Billing Internal (OP)

Before Allevo 4.0, cost elements of type 21 were posted via allocation type PC = primary costs, although these cost elements are actually secondary. At the latest when NO_ZERO_DELTA was used, this could lead to problems.

In current Allevo versions, AllocationType OP (= Order Settlements Primary) can be used for cost element type 21. All other settings are to be performed as for PC.

3.4.4 Direct Activity Allocation (DA)

Activity inputs of cost centers, orders and WBS elements

Constants	Cost center	Order	WBS element
(FD_)ALLOC_TYPE	"DA" or "2"		
FD_SENDER	ID sending cost center Mapping can be performed with the standard field COBJECT in the satellite or alternatively with a custom field in the append of the table (see section 3.5.9).		
(FD_)SND_ACT_TYPE	Activity type of sending cost center		
(FD_)OBJECT_TYPE	"KS"	"OR"	"PR"
FD_RECEIVER	ID cost center	Order	WBS element
(FD_)RCV_ACT_TYPE	Activity type for activity-type dependent planning		
FD_KSTAR	Cost element (only relevant for reading function in FP Read)		
FD_QNTY_FIX	Quantity (fixed)		
(FD_)DIST_KEY_FIX	Distribution key (fixed)		
FD_QNTY_VAR	Variable quantity, only for activity-type dependent planning		
(FD_)DIST_KEY_VAR	Variable distribution key, only for activity-type dependent planning		
(FD_)PLANYEAR	Main planning year		
(FD_)PLANVERSN	Plan version		
(FD_)PLANKOKRS	Plan controlling area		



Special Notes:

- If activity planning for a sending cost center (in the field FD_SENDER) is carried out differently from the initial object and FP is actively included in Allevo main planning, then the constant DUPLICATE_INCL should also be activated (see F1 help).
- Constant SIGN_MINUS can be used.
- Allevo also supports the posting of quantities for primary costs if "record quantity" is set in the master record for the cost element. The quantity to be posted is again expected in FD_QNTY_FIX / FD_QNTY_VAR.

3.4.5 Secondary Order Settlement (OS)

Indirect activity input through order settlement costs of cost centers, specifying the settlement cost element, sending cost centers and sender activity type.

Constants	Cost center	Order	WBS element
(FD_)ALLOC_TYPE	"OS" or "4"		
FD_SENDER	ID sending cost center The constant can be mapped with the standard field COBJECT in the satellite or alternatively with a custom field in the append (see section 3.5.9).		
(FD_)SND_ACT_TYPE	Activity type of the sending cost center		
(FD_)OBJECT_TYPE	"KS"	"OR"	"PR"
FD_RECEIVER	ID cost center	Order	WBS element
(FD_)RCV_ACT_TYPE	Activity type for activity-type dependent planning		
FD_KSTAR	Cost element (type 21)		
FD_QNTY_FIX	Quantity (fixed)		
(FD_)DIST_KEY_FIX	Distribution key (fixed)		
FD_QNTY_VAR	Variable quantity, only for activity-type dependent planning		
(FD_)DIST_KEY_VAR	Variable distribution key, only for activity-type dependent planning		
(FD_)PLANYEAR	Main planning year		
(FD_)PLANVERSN	Plan version		
(FD_)PLANKOKRS	Plan controlling area		

If activity planning is performed for a sending cost center (field FD_SENDER) different from the start object and if Flexible Planning is actively integrated in Allevo, the constant DUPLICATE_INCL should be activated (see F1 documentation).

Note: Constant SIGN_MINUS can be used.



3.4.6 Statistical Key Figures (SK, KF)

Constants	Cost center	Order	WBS element
(FD_)ALLOC_TYPE	Either "SK" or "KF" (same meaning) or "6".		
(FD_)OBJECT_TYPE	"KS"	"OR"	"PR"
FD_RECEIVER	ID cost center	Order	WBS element
(FD_)RCV_ACT_TYPE	Activity type for activity type-dependent planning		
(FD_)STAGR	Statistical key figure		
FD_QNTY_FIX	Quantity		
(FD_)DIST_KEY_FIX	Distribution key		
FD_QNTY_VAR	Maximum quantity		
(FD_)DIST_KEY_VAR	Distribution key		
(FD_)PLANYEAR	Main planning year		
(FD_)PLANVERSN	Plan version		

3.4.7 Budget and Total Cost Planning (BP, BD)

This feature allows budgeting and planning total costs independently of cost elements for orders (OR), projects (PR), investment programs (IP) and appropriation requests (AR). For projects, planning of total revenues independently of cost elements is additionally transferred. Equivalent SAP transactions:

- CJ40 / KO12 for cost planning (PR/OR)
- CJ30 / KO22 for planning original budget (PR/OR), CJ36 / KO24 for addendum.
- IM35 Posting plan values to position in investment program (IP)
- IM52 Budgeting of orders/WBS elements with assigned IM position
- IMA12 Changing appropriation requests (AR)

The following table shows the relevant constants.

Constants	Order	WBS element
(FD_)ALLOC_TYPE	"BP", "BD" (or "8") with following meaning: <ul style="list-style-type: none"> ▪ Both entries are equivalent for object type OR and PR ▪ For object type IP, "BP" is relevant for value type PLAN (= plan data) and "BD" for value type BUDG (= budget). 	
OBJECT_TYPE FD_OBJECT_TYPE	Previously supported object types: <ul style="list-style-type: none"> ▪ "OR" = internal orders 	



Constants	Order	WBS element
	<ul style="list-style-type: none"> "PR" = WBS element "IP" = Investment Planning "AR" = Appropriation Request 	
FD_RECEIVER	<p>For <i>OBJECT_TYPE = OR</i>:</p> <p>Order with one of the following order types:</p> <ul style="list-style-type: none"> 01 Internal order 02 Results-analysis order For budget values plus 30 (service and maintenance order) <p>For <i>OBJECT_TYPE = PR</i>:</p> <ul style="list-style-type: none"> WBS element (PSPNR or POSID) <p>For <i>OBJECT_TYPE = IP</i>:</p> <ul style="list-style-type: none"> Position in Invest program (see also constant IP_PROGRAM) 	
BUDGET_ACTIVITY	<p>"Value from" describes the activity (relevant object type in brackets):</p> <ul style="list-style-type: none"> KBUD = original budget (OR,PR) such as CJ30/CJ32, KBNO = Supplement (OR,PR) same as CJ37, KBRO = Return (OR,PR) like CJ38, KBFR = Release on Projects (PR), KSTP = Total Costs Planning (OR, PR) same as CJ40, KSTR = Revenues from projects (PR,AR) such as CJ42. <p>Additionally for WBS elements (PR):</p> <ul style="list-style-type: none"> Entry "X" at "Value to" activates the roll up of a cost planning within a project hierarchy (see details below or F1 documentation for the constant). <p>Action Request (AR):</p> <ul style="list-style-type: none"> OVCO = Overhead Costs (AR) IVCO = Investment Costs (AR) KSTR = revenues / annual plan total of revenues (AR) (thus corresponds to REVENUE). Does not exist as a total. <p>In "Value to" of BUDGET_ACTIVITY, a planning variant can be entered (alternatively, it can be determined using the constant (FD_)IP_APPROVALYEAR).</p>	
PLANYEAR	<p>Planning Year Information:</p> <p>"Value from" = planning year, for IP the entry "0000" causes a total value to be written instead of changes for a certain fiscal year.</p>	



Constants	Order	WBS element
		<ul style="list-style-type: none"> "Value to" = X means total budget (otherwise annual value).
ACTIVATE_AVAIL		Background activation of availability control: X means activation (for projects)
CHECK_PLAN_DATA		Consistency check in overall planning: An availability control is usually carried out if plan values are being transferred. Especially for total planning (activity KSTP and KSTR), a consistency check may additionally be performed: "Value from" in constant = X (replaces former constant NO_DATA_CHECK).
IP_PROGRAM FD_IP_PROGRAM		Name of the investment program (only relevant if OBJECT_TYPE = "IP"); A transfer from satellite is only possible from Allevo 3.5; previously, only one investment program was provided per FP scheme.
IP_APPROVALYEAR FD_IP_APPROVALYEAR		Approval year <ul style="list-style-type: none"> For OBJECT_TYPE = "IP" (Investment Planning): Transfer from satellite only possible from Allevo 3.5; previously, only one approval year was provided per FP scheme. For OBJECT_TYPE = "AR" (Appropriation Request): Finding planning variant via year and version (see notes below)
IM_DISTRIBUTION		Budgeting of orders/WBS elements with assigned IM position (see detailed notes below).

The following features are to be observed for **budget planning**:

- If the value in the field FD_VALUE_FIX shall contain a total budget (instead of an annual budget), "X" must be entered for the constant PLANYEAR (before: YEAR_INITIAL) in the column "Value to".
- The parameters FD_VALUE_FIX_y, YEAR_y and VERSION_y (y from 0 to 9) for additional annual budgets must always be included as unit in the constants.
- Budget values are checked in two aspects: budget totals of sub-projects may not exceed the budget of the superior element; the sum of all annual budgets may not be higher than the total budget.
- Postings can only be made within the time frame defined for the relevant budget profile.
- Background activation of the availability control for budgeted projects via the parameter ACTIVATE_AVAIL requires the activation type "2" to be set in the used budget profile.
- Supplements and returns are always posted as delta values.

Budgeting of orders/WBS elements with assigned IP position

Constant IM_DISTRIBUTION allows budgeting of orders/WBS elements associated with an investment program position that are normally budgeted from that IP position via budget distribution (equivalent to transaction IM52 or IM42, IM40, IM48). Corresponding settings:



- An X in column "Value from" activates the posting to IM position. Otherwise (indicator is not set) such orders/WBS elements will not receive budget.

If the indicator is set, but more budget is distributed to subordinate orders or WBS elements than existing on the corresponding IP program position, a warning appears in the log (no error message).

- Optionally, an X in column "Value to" activates the possibility to post a supplement (process KBNO, Supplement) or a return (process KBRO, Return). By default (without an entry), Allevo posts an initial budget (process KBUD).

The relevant operation must also be provided via BUDGET_ACTIVITY.

This feature is only available for object type OR and PR; as allocation type 'BD' must be specified.

Note:

For budgeting on assigned IP positions via constant IM_DISTRIBUTION Allevo uses the SAP function KBPP_EXTERN_UPDATE_CO with an extension that is only available in newer SAP systems. The prerequisites are described in SAP Note 2255427 (the corresponding correction must be imported if necessary). Please contact Kern AG support in advance for further clarification.

Further information in SAP Note 2249740.

Roll up/total up cost planning for projects

SAP provides various options to transfer cost planning content to higher levels within the project structure ("roll up"). Examples:

1. For transaction CJ30 and CJ40 the manual roll up of the values via menu "Edit >> Total up" can be used.
2. If the switch "Bottom-Up Planning" is active in the SAP planning profile, the calculation occurs directly when saving a new cost planning (CJ40 only).

Roll-up features within the project structure are also available in Allevo FP and correspond to the above-mentioned switch in the planning profile. Please note:

- Original budget cannot be rolled up, see constant BUDGET_ACTIVITY.
- The FP functions are based on function block KBPP_EXTERN_UPDATE_CO: Therefore, please observe SAP notes for this block (e.g. note 625613 with general notes or 1528776).
- In individual cases - especially if budgets with value 0 are partially transferred - the SAP module does not execute the roll-up completely. In this case a workaround via the constants BUD_POSTING_TYPE and BUD_POSTING_LOOP can help.

Technical realization

Allevo uses the following SAP interfaces/function modules:

- For investment programs (object type IP), Allevo uses the FB BAPI_EXPENDITUREPROGTREE_CHVAL. Further notes on the application are provided there. This



function block posts via delta values: Allevo reads the existing values in an upstream step (as in transaction IM36) and then posts the deviations.

- Allevo uses the FB BAPI_APPREQUEST_SETPLANVALUES for appropriation requests. Further notes on the application are provided there (e.g. on authorizations)
- Currently no BAPI for budget and total cost planning is available for OR and PR. As a result, Allevo uses the SAP function module KBPP_EXTERN_UPDATE_CO according to SAP Note 625613 = "Interface for overall planning and budgeting". The SAP note contains additional specifications, e.g. for updating line items, budget check, availability control, roll-up of budgets.

For the application of constant IM_DISTRIBUTION for budgeting orders/WBS elements that are linked to an investment program position, SAP Note 2255427 are to be observed and the corresponding correction must be imported if necessary. Authorizations equivalent to IM52 should be assigned to the user.

Example of planning in the investment program (IP):

- OBJECT_TYPE is "IP" (Investment Planning); FD_ALLOC_TYPE is "BP".
- IP_PROGRAM contains the name of the investment program, FD_RECEIVER the relevant position in the program.
- The approval year permanently stored in the FP scheme via IP_APPROVALYEAR.
- Via FD_PLANYEAR 0000 is transferred as fiscal year in order to allow posting as total value. For planning (BP) a version should be transferred additionally (thus e.g. in FD_PLANVERSION).
- The total value of investment is stored in FD_VALUE_FIX.
- The constants FD_VALUE_FIX0 FD_VALUE_FIX2 contain the plan values of the first three years. During program execution, all values stored here are being written. COLDEF0 to COLDEF2 should be set for the assignment of year and version via an Allevo-TimeSet.

Example of planning for appropriation request

- OBJECT_TYPE is AR (= appropriation request); FD_ALLOC_TYPE is BP
- BUDGET_ACTIVITY contains in "Value from" the operation type (e.g. IVCO) and in "Value to" the plan version (also see note below).
- FD_VALUE_FIX contains the field with the plan value from Excel
- When planning total value, an X must be entered in "Value to" of PLANYEAR

An appropriation request can be planned using a planning variant:

- The variant can be entered in column "Value" to of BUDGET_ACTIVITY.
- Alternatively, in current Allevo versions, the appropriate planning variant is determined automatically via approval year and version if the constant FD_IP_APPROVALYEAR or IP_APPROVALYEAR is set up (see assignments in the master record for appropriation request; or table IMAVZ). This determination overrides the entry at BUDGET_ACTIVITY.



The variant determination is always executed: If neither option is successful, an error message appears in the log.

Earlier program versions required all relevant information in one row of the satellite (with unwinding over two schemes); current program versions additionally allow sequential display of annual values in rows. See example:

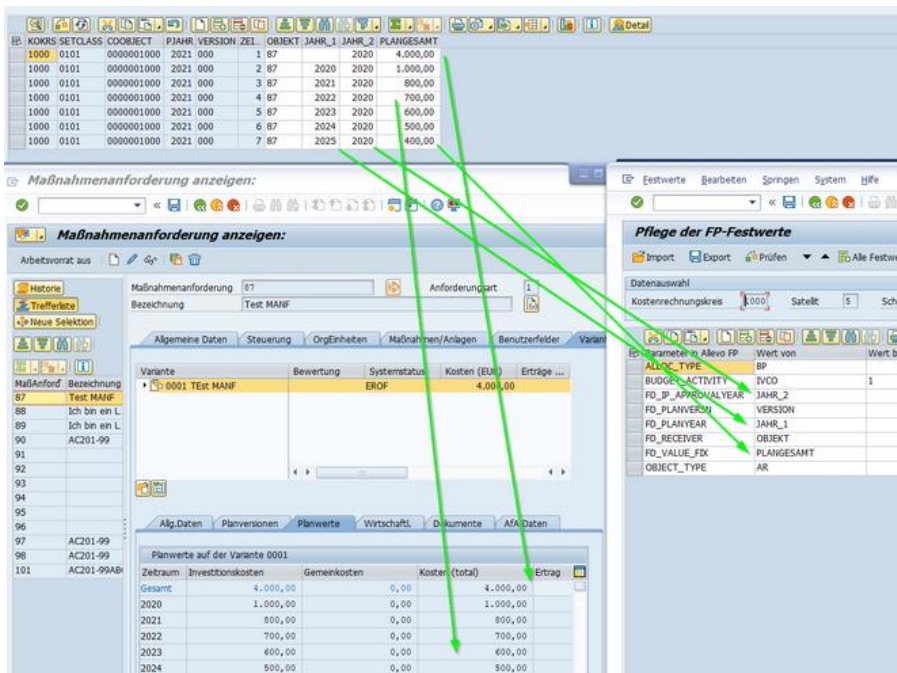


Figure 3.2 Satellite table and mapping to post budget for appropriation request

At the same time, existing plan values (total or annual values) are retained if no values are stored for them in the new plan data.

Read budget data

Allevo FP does not offer any menu functions for reading budgets, a general function module /KERN/IPP_READ_BUDGET is available, which can be included in customer-specific processes. Please involve Kern AG support for further clarification.

Alternatively, a total budget (OR, PR) can be read in Allevo main planning via constant MAP_FIELDxx.

3.4.8 Profit Center Planning

Direct profit center planning may be performed both for the classical and the new general ledger depending on the setting of the Allevo constant PC_NGLA_PLANNING: Depending on this, Allevo uses either the batch input procedure (e.g. for 7KE1, GP12N) or BAPI call (new general ledger). Using the New General Ledger, the table /KERN/IPP_PC_NGL must also be maintained via transaction /ALLEVO/PC_NGL_MNT, so that the relevant characteristics are also transferred.

Detailed explanations on profit center planning can be found in the Allevo manual “Profit Center“.

In the following, only constants required for Flexible Planning will be described.



Constant (FD_)ALLOC_TYPE controls the characteristic of a posting:

- AC / 10: Simple posting with account
- PP / 11: Posting to account & partner profit center
- FA / 12: Posting to account, partner profit center and functional area
- PT / 13: Posting to Account, Partner Profit Center, Functional Area & Partner Company
- AA / 14: Posting to a balance sheet account
- SK, KF or 6: Posting to a statistical key figure

Constants	Profit Center
(FD_)ALLOC_TYPE	"AC", "PP", "FA", "PT", "AA", "SK" or "KF" (see numerical values above)
(FD_)OBJECT_TYPE	"PC"
FD_RECEIVER	Profit center (receiver)
(FD_)RCV_CCODE	Company code (receiver)
(FD_)KSTAR	Account
FD_SENDER	Sending Profit Center (usually in COBJECT)
(FD_)SND_CCODE	Company code of the partner profit center (sender)
FD_FUNCTIONAL_AREA	Functional area (for Alloc.Type FA or PT)
FD_TRADING_PARTNER	Number of the trading partner (only for Alloc.Type PT)
FD_QNTY_FIX FD_QNTY_VAR	Quantity fixed or variable: Sum if both fields have a mapping <i>Note on the compatibility with former versions: The content of FD_QNTY_VAR is interpreted as value as long as FD_VALUE_FIX/FD_VALUE_VAR is without mapping</i>
FD_VALUE_FIX FD_VALUE_VAR	Value fixed or variable (the sum will be calculated if both fields have a mapping)
INVERSE_ACCOUNTING	Automatic offsetting entry to the partner profit center (reverse entry). Is also taken into account when cancelling.

For the New General Ledger, Flexible Planning can adopt the ledger from the Allevo constant of the same name (standard is "OL"); for the Classic General Ledger, planning is always performed via "8A".

3.4.9 Balance Sheet Planning via FP

From version 3.5.8 FP also supports balance sheet planning for classical and new general ledger. A suitable Allevo layout must be assigned in order to evaluate the relevant constants (for details on the settings, see the Allevo manual "Profit Center"):

- Posting to New General Ledger, the Allevo constant PC_NGLPC_READTABLE is activated.



- For classical general ledger PC_READTABLE is evaluated (for balance sheet planning "GLT0" must be entered there).

The field mapping (constants) is to be set up equivalent to profit center planning, e.g. "PC" as OBJECT_TYPE. Constant FD_RECEIVER has no meaning in this case; it is reasonable to assign the company code here (i.e. the same mapping as for FD_RCV_CCODE).

3.4.10 Allevo Objects (object type KX)

Flexible Planning is able to use the Allevo object as start object in order to summarize e.g. different postings according to superior criteria in a common posting run.

Furthermore, postings can also be made for the Allevo object: All relevant plan data are stored in an Allevo own database table (/KERN/ACOSx). The relevant basic functions are described the "Allevo SAP manual".

The following allocation types are available to generate such postings via Flexible Planning:

- AP / 15 = Plan primary costs (ACOSP)
- AF / 16 = Actual primary costs (ACOSP)
- AS / 17 = Secondary costs (ACOSS)
- AK / 18 = Statistical key figures (ACOSR)

Other general constants are to be used in the same way as for the original SAP posting procedures (e.g. PC and SK/KF).

Optionally, certain organizational levels can also be specified. Example for the functional area in the field FD_FUNCTIONAL_AREA: It's possible to enter the corresponding abbreviation into the table field containing the origin in order to obtain unique rows in the database table (for details see F1 documentation for the constant).

Special function for posting "actual data"

Usually the ACOSx table is used to enter plan data that cannot be mapped using the SAP standard plan functions (e.g. to simulate objects or cost elements).

For specific use cases it can also be useful to use ACOSx to simulate actual data (e.g. transfer from external systems or legacy data transfer). Advantage: All Allevo standard functions can then be applied to the external data (e.g. reading reference data with selection via year/version).

For collecting primary "actual data" FP offers the allocation type AF (= Allevo Primary Costs ACOSP Actual data) with the same functions as for AP. See also the corresponding section in the "Allevo SAP manual".

3.5 Other Program Settings

3.5.1 Multi-Year Planning (subsequent years in one row)

The constants (FD_)PLANYEAR, (FD_)PLANVERSION and (FD_)PLANKOKRS are available for all allocation types. Planning can thus be performed row by row over many years.



Often, however, multi-year planning is desired, whereby the input of annual values and quantities shall be entered over several columns of a single Excel row. FP supports this use case with up to 10 years that can be processed in parallel.

This feature requires additional fields in the satellite for the fixed and, if needed, variable quantities/values. The following parameters are available for the constants (entry in column "Value from"):

Parameter	Assignment of satellite field names
FD_QNTY_FIX0 ... FD_QNTY_FIX9 or FD_VALUE_FIX0 ... FD_VALUE_FIX9	<p>Mandatory parameters in multi-year planning</p> <p>Name of the fields showing the (fixed) quantities or values for the years 0-9: See section 3.5.1 and documentation for the constant FD_VALUE_FIXy.</p> <p>Alternatively, these constants support annual planning across different cost elements within a single satellite row (see section 3.5.2).</p>
FD_QNTY_VAR0 ... FD_QNTY_VAR9 or FD_VALUE_VAR0 ... FD_VALUE_VAR9	<p>Optional parameters in multi-year planning</p> <p>Name of the fields showing the variable quantities or values for the years 0-9.</p>
YEAR0 to YEAR9	Year for which quantities are to be posted: Either fixed as a four-digit number (e.g. "2015") or relative to the main planning year (e.g. "+1"). The relation is recognized automatically by the sign (available from Allevo version 3.3.29).
VERSION0 to VERSION9	Version for which quantities are to be posted.
COLDEF0 to COLDEF9	<p>Allevo-TimeSet for determining year and version.</p> <p>Important: The multi-year planning always refers to complete years. Posting is therefore performed over periods 1-12 (or set special periods), regardless of related entries in the TimeSet.</p> <p>These parameters allow referring to requirements in the Allevo layout. They will only be considered if year and version are not defined by the previously mentioned parameters.</p>

The parameters FD_QNTY_FIXy or FD_VALUE_FIXy and YEARY (y from 0 to 9) must be maintained as a unit in the constants. Without any entry concerning the version (e.g. via the constant VERSIONy), the version of the main planning year will be used.

The feature is only available for planning on an annual basis. The allocation types DA, OS, PC and BP are supported. The Reading Mode is not available.

Note:	Detailed information about the settings for this use case can be found in the F1 documentation for the constant FD_VALUE_FIXy.
	Alternatively, the use of the fields (FD_)PLANYEAR, (FD_)PLANVERSION, and (FD_)PLANKOKRS allows unrestricted row-by-row planning over multiple years.
	When planning more than 10 years in parallel, a second FP scheme with mapping to the values in the remaining years can be helpful. To enable updating directly via Allevo main planning, both schemes may have to be called simultaneously via *.



3.5.2 Planning with Different Cost Elements in one Row

The constants of the multi-year planning (FD_QNTY_FIXy, FD_VALUE_FIXy) can also be used for a different use cases as explained in the example in the following table.

Order number	Text	Cost element 465000 Gasoline	Cost element 466000 Tax	Cost element 467000 Tolls
FR-SD_456	Service vehicle	4.500,00	460.00	500.00
FR-SE_348	Control vehicle	7,800.00	460.00	650.00

The table shows data of a fleet planning; plan values are displayed in columns next to each other: Each column is assigned to a certain cost element.

In order to use this feature, Flexible Planning must know the relevant cost element. This allocation is performed via "Value to" of constants FD_QNTY_FIXy (y from 0 to 9) or FD_VALUE_FIXy: the cost element is to be entered here.

The parameters FD_VALUE_FIXy (or FD_QNTY_FIXy) and YEARy must also be stored as a unit in the constant in this use case. Up to ten different cost elements may therefore be addressed, or for example five cost elements distributed to two years. Otherwise, notes as in the chapter on multi-year planning apply (see above).

Note:	If "Value to" is not specified, FP will adopt the entry from constant (FD_)KSTAR, as in multi-year planning.
	Detailed information about the settings for this use case can be found in the F1 documentation for the constant FD_VALUE_FIXy.

The described feature for planning over several cost elements is, for example, useful if value fields in Allevo COPA planning shall be posted to suitable cost elements at the same time.

3.5.3 Period Planning

FP allows the blockwise processing of period values via the so-called "period/monthly planning": In this case, the periods are mapped via twelve columns in the satellite.

Necessary prerequisites:

- The feature is activated via constant MONTH_ACTIVE with "X" in column "Value from".
- For the constants FD_QNTY_MNTH_FIX and FD_QNTY_MNTH_VAR, the quantity fields for months must be stored in the satellite (or FD_QNTY_FIX and FD_QNTY_VAR in case of years).
- The value fields for months are to be stored for the constants FD_VALUE_MNTH_FIX and FD_VALUE_MNTH_VAR (or FD_VALUE_FIX and FD_VALUE_VAR in case of years).

In the mentioned constants, the name of the periodic quantity/value satellite fields must be specified without the two end digits (the end digits stand for the periods 01 to 12; special periods can also be possible according to SAP Customizing).

When using multiple period blocks, it has proven useful to use field names in the satellite by which the respective block can be recognized (e.g. BLOCK1xx, BLOCK2xx). In any case, the individual fields per block must be numbered with xx = 01 to 12 at the end: It is NOT possible to process fields with "01" to "24" like two blocks.



Note:	Not in all use cases, all 12 periods are actually used for the entry of plan data and possibly not all corresponding fields are created in the satellite. In this case, actually used periods must be entered in "Value to" for MONTH_ACTIVE (separated by comma). Otherwise, FP expects the satellite fields for all 12 periods.
	The period planning can either be done for one fiscal year and combined with the multi-year planning (in one row; see section 3.5.1) or expanded to multiple years by using the field parameter FD_PLANYEAR.

3.5.4 Posting of Individual Monthly Values or Ranges (from/to)

In individual cases of annual planning it may also be useful to restrict the plan values to individual periods or period ranges, e.g. posting for a single month or quarter. Plan values outside the period range retain their old value.

The function is activated via the constant PLANPERIOD with fixed defaults, or alternatively via FD_PLANPERIODFROM / FD_PLANPERIODTO if the number of the respective from/to period is stored directly in the satellite.

This function can also be useful if period values are sequentially in a satellite and not in columns (e.g. twelve rows for a period block): FP may then generate 12 individual transactions to post a complete year.

3.5.5 Special Case: Periods in Rows

Part of the flexibility of FP is to transfer data with different source structures in a common posting process to SAP. FP handles a special case when a "Z" is entered for the constant MONTH_ACTIVE:

- In this case, satellite data where each period has its own row can be processed (FP usually expects period data in columns).
- The special function reads the data in the block, aggregates internally to one row with all periods and then transfers to SAP in one operation.

Without this special function for the constant MONTH_ACTIVE, the postings could also be performed via suitable mapping of the constants FD_PLANPERIODFROM and FD_PLANPERIODTO (entry of the period column): But this would lead to a separate SAP line item document for each period.

3.5.6 Special Periods (13-16)

In both annual planning and period planning, Allevo FP supports the use of special periods (13-16). For this, the number of relevant periods is taken from the fiscal year variant assigned to the current controlling area (table TKA01 and T009). A corresponding customizing must exist on the SAP side, otherwise the standard case with 12 periods/months applies.

3.5.7 Currencies

The constant CURRENCY_TYPE defines the currency for planning:

- C: Controlling area currency (default)
- O: Object currency
- T: Transaction currency



In the last case (T = transaction currency), the currency key must additionally be transferred as ISO code (3 digits); either:

- in the field "Value to" of CURRENCY_TYPE or via
- FD_CURRENCY with mapping to a satellite field.

Example: Plan data shall be entered in different currencies for intercompany planning (e.g. debit in English pounds, credit in euros). In this case, the posting must be performed using the transaction currency of the mapping in FD_CURRENCY.

If the constant CURRENCY_TYPE is not set, FP can alternatively take the currency type from an Allevo TimeSet (see constant COLDEF).

3.5.8 Delta Planning

In the standard case, Flexible Planning interprets the entered plan value as total value. Optionally, you can use the constant DELTAPLANNING to plan delta values and quantities. An application example would be the planning of transfer postings between cost centers.

Activation may also be performed via the corresponding setting in the Allevo-TimeSets.

Important: Delta postings create the risk of posting multiple entries (e.g. by double posting of satellite data when executing Flexible Planning twice). Appropriate arrangements are to be made in the project flow.
See detailed documentation on the constant.

3.5.9 Sender Selection for Activity Allocation and Sec. Order Settlement

Usually, in activity planning via Allevo FP, the activities provided by the respective cost center are recorded (e.g. for other cost centers): This is therefore from the view of the "sender". It means in the following for working with FP:

- The corresponding receiver is stored in a satellite field addressed via the constant FD_RECEIVER.
- The sending cost center is stored in the key field COBJECT of the satellite; usually the initial object of Allevo planning.

However, the planning process sometimes requires a planning cost center to specify a different cost center as sender of an activity type. In this case, plan data are recorded in the satellite as data record of the planning cost center, but the allocation of activities is done for a different sender-recipient relation.

For this purpose, the satellite must provide another field specifying the new sender. The field name is to be entered in the constant FD_SENDER.). If a satellite data record now specifies a cost center in this field, it is used as the sender for posting. If the field is not available or empty, the sender defined in the key field COBJECT is used (prerequisite: a cost center type object is available).



3.6 Cancellation Feature for Changed or Deleted Rows

3.6.1 Background and Properties

The functionality of Flexible Planning to address data records in the satellites allows high flexibility but also bears certain risks. What happens to data records in the satellite that are completely deleted or whose key data are modified after posting data in SAP?

Goal: If the user adjusts data in Excel, he/she also expects full adjustment of the respective data in SAP. This means that not only the new data record must be posted but that adjusting the posting for the old data record may also be required.

This functionality is implemented by a cancellation feature running in the background: For this, the constant CANCEL_ACTIVE must be active ("X" in column "Value from") in the relevant execution scheme.

Some characteristics:

- The cancellation feature is implemented for all allocation types including all posting options for profit center planning. All quantity and value fields for a posting record are set to zero for the corresponding allocation types and the distribution key is set to 2.
- Special features like multi-year planning (see section 3.5.1), planning over several cost elements (see section 3.5.2) and flexible sender selection (see section 3.5.9) are taken into account.
- The cancellation features also support satellites that are included via constant SATELLITE_INCL.

Important: Modifications of the mapping/scheme may only be made if the complete cancellation list is processed. Otherwise, there would be a high risk of error postings of data.

3.6.2 Cancellation Procedure

The cancellation is processed in two steps:

1. The data records to be cancelled are transferred to a "cancellation workbook" (table /KERN/IPP_CANCEL). All key data of a posting are stored there.

Restriction: Storage in the cancellation table only takes place if the changes to the satellite contents are made via the Excel integration in the Allevo planning transactions (Allevo Workplace) or in the shuttle (Layout 1). The cancellation feature, on the other hand, does **not** work when manually editing data in a satellite using other tools.

2. All data records in the "Cancellation workbook" are processed the next time the FP planning features are called up; i.e. all existing SAP postings are set to the value "zero".

The cancellation process can be executed both automatically in the Allevo Workplace and in isolation (thus with a time delay) via the FP dialog transaction. See also section 4.5 on the constant ALLEVO_ACTIVE.

FP starts the actual cancellation in SAP (postings via BAPI's with the zero values and quantities) **before** the update of new data records. The pivot of the cancellation process is the table /KERN/IPP_CANCEL. A cancellation record is saved to this table exactly when at least one key satellite field has been changed or an entire row has been deleted. In case of a set up multi-year planning, exactly one cancellation is made for each year.



The cancellation routine runs completely in the background; thus there are no "success messages". Possible error messages of the SAP BAPI's are displayed to the user.

3.6.3 Cancellation Table

As in the default behavior of Allevo FP, the cancellation table is searched for data records to be cancelled via the start object (i.e. based on the fields Setclass and CO-Object) or via the receiving object (i.e. via Object_Type and Receiver) and of course also depending on the controlling area, satellite, schema, year and version.

The current processing status of the cancellation records in table /KERN/IPP_CANCEL is reported in the "Status field", which may contain the following values:

Value	Description
0	To be cancelled
1	Cancelled
2	Faulty (could not be cancelled)
3	Deferred (no cancellation routine for this allocation type).
4	Deactivated (cancellation no longer activated)

The current contents of the cancellation table are displayed via button |Cancellation list| in transaction /ALLEVO/FP: Optionally, a manual change to the status in order to start a new processing is also possible there.

The status in the cancellation process:

- In the first step, cancellation-relevant data are entered in the table with status 0. In the standard case, only these data records are considered in the subsequent cancellation process.
- After a successful cancellation, the status is set to 1.
- Status 2 indicates that the BAPI transactions report an error during the cancellation in SAP (this error message is displayed during program execution).
- Status 4 stands for data records that have been entered in the cancellation table in a first step but that the user has deactivated before executing the second step.

If an error occurs during execution of the cancellation routine because stored data cannot be processed in SAP, FP will ignore this cancellation record in all future calls. However, a different behavior can also be set via the constant CANCEL_ACTIVE.



4 The Planning Mode

4.1 Execution Options

The dialog mode is started by Flexible Planning via transaction /ALLEVO/FP or the Allevo Cockpit (the direct call from an Allevo planning transaction will be described below).

Allevo: Flexible Planung (FP)	
	Schemata Festwerte Stornomappe Allevo Version
Datenauswahl über	
Kostenrechnungskreis	1000
Satellit	01 Invest Planung über Allevo
Schema	DSAG ABSCHREIBUNG KOSTENSTELLE
Planjahr und -version	
Geschäftsjahr	2017
Version	0
Objektauswahl	
Objektart	KS Kostenstelle
Kostenstelle	1000 bis <input type="text"/>
Setname	<input type="text"/>
Objektauswahl über	
<input checked="" type="radio"/> Sender (EinstiegObj. COBJECT) <input type="radio"/> Empfänger (Festw. FD_RECEIVER)	
Ausgabe	
Ausgabe Level	<input type="text"/>
<input type="checkbox"/> Vorschau Satellitendaten <input type="checkbox"/> Vorschau Buchungsdaten	

Figure 4.1: Start in Planning Mode

The first selection criteria to be entered are the controlling area, the satellite and the appropriate execution scheme. With an entry * in the field "Scheme", all schemes stored for the satellite will be executed in one program run (see special notes in the following section).

The other parameters determine the content to be processed:

- The entries for "fiscal year" and "version" are selection criteria only regarding the entries in the satellite (supplementary to the controlling area). Normally, they do NOT determine which planning year and version the posting receives: This is determined by the relevant constants such as PLANYEAR and PLANVERSN or the corresponding constants in the mapping. Only when both parameters are not defined via a constant, "fiscal year" and "version" are used as criteria for posting.
- The entry for the object type defines further selection fields, for example cost centers and cost center group for object type KS. Upon entry of a group or a From/To range, the included objects will be determined at a time defined by the fiscal year in the start screen. This can differ from the main planning year (depending on the settings in the constants such as PLANYEAR).
- A further restriction when reading satellite data can result from combining the constants FD_ALLOC_TYPE and ALLOC_TYPE in the mapping: In this case, only data records with the allocation type in field FD_ALLOC_TYPE corresponding to the entry in ALLOC_TYPE will be processed.



With the help of two further options, it is possible to decide if data shall be processed from a sender or receiver perspective:

- **"Sender (in COBJECT)"**

In most FP use cases, plan data for recording and processing costs is entered from the sender perspective (who performs the activity): This sender (e.g. a cost center) is in the field COBJECT.

When selecting and processing the satellite data for the sender via the field COBJECT, the field SETCLASS is also considered according to the current object type. This also applies if a different table field was determined as sender information via the constant FD_SENDER, for which the actual posting then takes place (see section 3.5.9).

To get started, using Allevo's own object type KX is also possible, for example, to process postings according to higher-level criteria in a common run.

- **"Receiver (in FD_RECEIVER)"**

In this case, Allevo selects FP via the satellite field that is stored as receiver in the constant FD_RECEIVER (and FD_OBJECT_TYPE for the object type). Unlike usual in Allevo, the fields COBJECT and SETCLASS in the satellite index are NOT relevant for this selection (only controlling area, year and version). Two application examples:

- (1) All costs should be posted for which order 1234567 is entered as the receiver (regardless of the cost centers that planned the activities).
- (2) Costs via different object types are collected (e.g. from cost centers and orders as "sender"). For the receiver, these costs should be posted in total, independent of the respective sender object type.

Additional Notes:

This option is only available in FP dialog mode. When calling directly from an Allevo planning transaction, the selection is always made via the field COBJECT in the satellite index (i.e. corresponding to the first option for the sender).

When executing the selection, Allevo FP simultaneously observes data with internal and external format of an object number: This is particularly helpful if the satellite's data was captured via Allevo, since the object number is not subject to automatic conversion in this case. In order to store in the shuttle with an external representation, the constant SHUTTLE_CONVERSION can be helpful.

If many data is stored in the satellite, it may be useful to create a separate database index for the relevant satellite fields.

Output options

The parameter "Output Level" defines with which details the user should receive messages after program execution (see section 0 below).

Allevo FP contains two ALV preview functions that are particularly useful for analyzing results when data from multiple satellites are posted together (e.g. when using the constant SATELLITE_INCL):

- The option "Preview Satellite Data" shows the rows selected by FP for processing (multiple satellites for SATELLITE_INCL are shown in separate screen).



- "Preview posting data" lists the data after processing the mapping in the current scheme and before transferring it to the posting module.

4.2 Running Multiple Schemes

Typically, for the execution a single scheme is provided in the initial screen of the relevant FP transaction (see FP initial screen in Figure 4.1). For the execution of multiple schemes, the following rules apply:

- Entry * in the field "Scheme" prevents all schemes stored for the satellite from being executed in one program run.
- Alternatively, "partial masking" can be performed using *: Entering IV*, all schemes starting with IV are executed.
- If no scheme is entered in the selection parameter, the transaction searches for all schemes with constant ALLEVO_ACTIVE being active (entry in "Value from").

Exception: Schemes with the component "READ" in their name are ignored in Planning Mode (resp. only considered in Reading Mode).

Note:

When calling directly from Allevo main planning via ACTIVE_FP, suitable schemes are being selected via the name of the calling Allevo layout. Only FP schemes for which this layout is entered in the FP constant ALLEVO_ACTIVE are executed (or if there is no restriction stored).

Execution takes place in the order of the stored item numbers per scheme: If the order of execution is relevant for the posting process, this must be considered when assigning the item numbers to the scheme.

If several schemes are executed in succession, individual control can also be useful for error handling: Depending on the setting, Allevo prevents the execution of subsequent steps as soon as an error has occurred in a preceding step (see constant SAVE_LOG).

4.3 Processing with Automatic Aggregation

Usually, satellite entries are much more detailed than necessary for posting in SAP. Allevo FP therefore tries to aggregate data via the relevant key fields before a posting is executed (i.e. before transfer to the SAP BAPI).

Example: The investment planning of a cost center plans 3 actions that later must be processed for 2 investment orders only. In this case, FP aggregates the values/quantities of the two data records to be posted to the same investment order.

4.4 Process Data from Multiple Satellites Together (SATELLITE_INCL)

Using the constant SATELLITE_INCL, FP is able to combine, aggregate and jointly process the data from several similarly structured satellites. When saving the satellite in Allevo, in this case, the sequence must be considered. For setup details, see F1 documentation on the constant.

Use case: Depreciation data for existing assets and new orders are kept separately but should be posted together.



4.5 Integration into Allevo Planning Transactions

It is often wanted that postings are made at the exact moment when the user executes the function |Adopt plan data| in the Allevo main program. For this use case, an automatic call must be installed. Allevo then directly transfers the required selection parameters (such as controlling area, plan year, version, cost center...) and starts Flexible Planning.

Two conditions must be met for the automatic call:

- On the side of Flexible Planning, the constant ALLEVO_ACTIVE must be active and the relevant Allevo layout must be entered in "Value to". Various entries for the layout are possible (separated by comma): Flexible Planning will then be executed only for the layouts mentioned here.
- The constant ACTIVE_FP must be set in the Allevo main module. A combination of numbers in the column "Value from" defines the satellites that data shall be posted for via Flexible Planning. The feature "Sat.Assistent" is recommended for the entry.

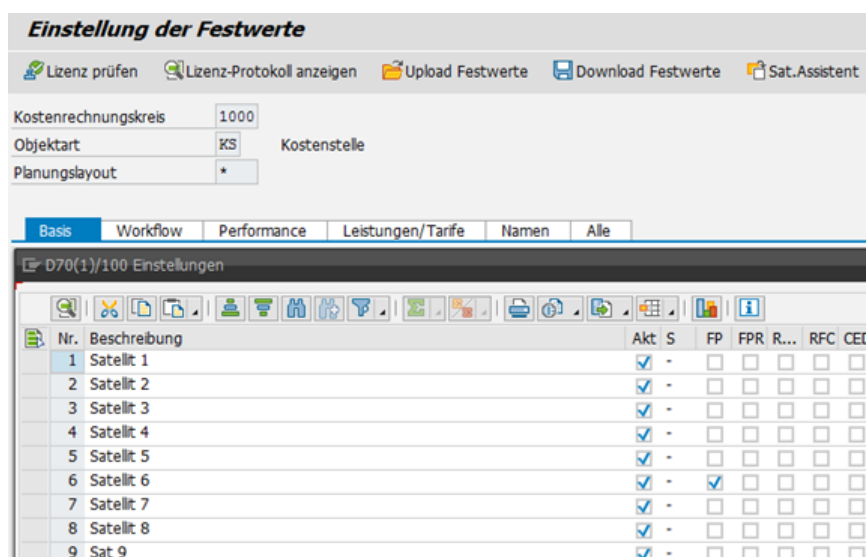


Figure 4.2: Activation of the Allevo constants for Flexible Planning (here satellite 6)

See the "Allevo SAP manual" or the F1 help for the constant ACTIVE_FP for further details.

Note: If Flexible Planning has NOT been activated on Allevo side, monitoring for cancellations will still be done as soon as a scheme has been created for the satellite (i.e. cancellation list will be maintained).
 Accessing Allevo constants is often required also when calling from Flexible Planning in the dialog. Flexible Planning will in this case use the first layout entered for ALLEVO_ACTIVE.

By default, the relevant data is selected via the field COBJECT of the satellite (i.e. after "Start object"). The constant SEL_AS_RECEIVER allows alternatively the selection via "receiver" (from version 3.4.4).

4.6 Background Processing for Performance Improvement

Making a large number of postings via FP can have an impact on performance (e.g. postings for profit centers). In this case, it can be advantageous to run Flexible Planning in the background.



This is possible via a permanently scheduled job for transaction /ALLEVO/FP (e.g. hourly, report name /KERN/IPPPF, the job is to be created via SM36) or alternatively via a specific call from Allevo.

Background mode when called from Allevo main planning

Controlling the direct call is done via the constant ACTIVE_FP_PARAM with entering BACKGROUND in column "Value from" (for details see F1 documentation on the constant). Characteristics:

- In this case, it does not wait for FP to finish execution but passes processing to SAP transaction SM58 (execution is also logged there).
- On FP side the Application Log should be active to evaluate FP messages in transaction SLG1 (see FP constant SAVE_LOG).
- If the postings are to be kept in different satellites and processed in succession, then the order in which they are processed can be relevant (e.g. if data that is included via the constant SATELLITE_INCL is to be saved before it is processed via the main satellite). Usually Allevo itself stores in the order of the satellite numbers; an individual order can be set via the Allevo constant ACTIVE_SAT_SORT.

Note:	From version 3.4 onwards, this sorting of satellites is adhered during processing in the background.
--------------	------------------------------------------------------------------------------------------------------

Error messages should be written to the SAP Application Log when executing in background mode (see e.g. constant SAVE_LOG).

4.7 Additional Authorizations / RFC Call

Depending on the use case, it can happen that a planner does not have all the authorizations required for posting (e.g. for posting with activity output; here, the user requires authorization for posting to the receiving cost center). In such cases, it may be useful to make the posting via an RFC call, where only the user in the RFC destination must be sufficiently authorized.

Controlling the direct call is done via the constant ACTIVE_FP_PARAM with entry of the RFC connection in column "Value from". For details see F1 documentation for the constant: The necessary authorizations for the RFC user are also described there.

If no direct call from Allevo is required, a central controller could execute the FP postings in dialog (i.e. call via transaction /ALLEOVO/FP) or schedule execution via a job (see background processing in the previous section).

4.8 Allevo Status Management

In order to write plan data via Flexible Planning, the corresponding planning object must have a valid entry in the Allevo status management. This ensures that Allevo license terms will be complied with also when planning with Flexible Planning.

In the standard case, Flexible Planning itself creates the status entry "3" for the first posting to a new planning object (e.g. for the first posting to a cost center):

- For a long time, the query was made for the year and version that is entered in the satellite, i.e. for which is currently to be posted. From Allevo 4.2, the main planning year and plan version that are stored in the corresponding Allevo layout are relevant for the status query (without a layout, it is the year from system date + 1).



- “FP” is always used as the name of the layout and as the abbreviation for the version. The constant STATUS_INFO allows setting a different abbreviation for layout and version (see explanation on the constant).

This procedure makes it possible to see whether the status information originates from Flexible Planning or Allevo.

Note: Layout “FP” and version “FP” do not obligatorily need to be created in the system. From Allevo 3.3, the status management will still display entries for this layout (as long as there are layouts available). Before, the status modified via Flexible Planning was preferably written into the same layout used also for starting Allevo; such a layout possibly had to be created also with a different object type.

With former program versions, it is recommended to create at least one additional Allevo layout for working with Flexible Planning (without parameters, constants ...) in order to recognize objects in the status management that had already been posted from there.

The Allevo license may be installed in the way that posting via Flexible Planning is possible also without the main license for Allevo.

4.9 Checks and Error Messages

When executing the program with the entered selection parameters, the following inspections are performed:

- The mapping of fields is checked for completeness; the execution of the scheme will be cancelled in case of errors.
- In case that the license for the current object type and controlling area is invalid, the relevant satellite rows will be excluded from planning (skipped).
- The Allevo planning status must be set with regard to the main planning year (same query as in the Allevo main module) for all posting objects.
- Incomplete satellite rows will be skipped and finally listed in the error log (e.g. in case of invalid entry of controlling area, allocation type or object type)

All messages are collected and will be displayed in the end of program execution or attached to the log upon call from Allevo. In the dialog mode, the specification for “Output level” or “Message level” defines at which level/importance messages of the program shall be displayed (e.g. only errors, see also Figure 4.1).

Via the constant SAVE_LOG, you may additionally set per scheme, if and at which level messages shall be saved in the SAP Application Log (transaction SLG1).

Flexible Planning creates a status message about the number of read satellite and posting rows transferred to the respective SAP-BAPI for each successfully completed scheme.



5 The Reading Mode (FP Read)

5.1 Overview and Initial Screen

The Reading Mode of FP is called via a separate transaction /ALLEVO/FPR: The transaction internally uses the same functions as the Planning Mode (initial screen, constant maintenance...). Therefore, selection variants for the initial screen are also displayed in both transactions.

Application areas in the Reading Mode:

- The Reading Mode was originally implemented to read activity inputs that have already been planned by the receiver (corresponds to reading secondary costs from table COSS). The aim is to display the "scheduled quantities" to the sender, breaking down the receiver information.
- Alternatively, from Allevo version 3.2, it is possible to read out activity inputs planned by the receiver via the Allevo module ProCED. The variant ProCED offers the advantage that all other planning features can easily be mapped via the Allevo standard. In new projects the implementation via ProCED should therefore be checked first.
- From Allevo version 3.5, FP Read also allows to read data on primary costs (i.e. table COSP) in order to possibly derive further planning from it (e.g. for balance sheet planning with aggregated data).

The order of the data records transferred to the satellite during readout cannot be influenced. The level of detail is predefined by the posting in SAP. Existing data is overwritten in the satellite.

Datenauswahl über	
Kostenrechnungskreis	1000
Satellit	00
Schema	READ_Y
FP Gesamtbudgetierung Kostenstellen Lesen Jahre	
Planjahr und -version	
Geschäftsjahr	2010
Version	0
Objektauswahl	
Objektart	KS Kostenstelle
Kostenstelle	1000 bis 1200
Gruppenname	
Ausgabe	
Ausgabe Level	I
<input checked="" type="checkbox"/> Vorschau Satellitendaten	

Figure 5.1: Call FP in Reading Mode

For the reading function, one execution scheme with its name containing the letters *READ* must exist (e.g. name READ_Y as in the example above). Allevo FP checks this name agreement during execution.

It is also possible to use several READ schemes per satellite (e.g. READ_Y, 1READ). Via entering * in the field "Scheme", all READ schemes of a satellite are processed in the specified order: In such cases, the read information does not overwrite each other, but is transferred to the satellite one below the other with a new row number.



It is controlled via the constant MONTH_ACTIVE and the corresponding FD_QNTY fields whether the data is read on a monthly or yearly basis and stored in the satellite (see section below).

5.2 Call from Allevo Workplace

The Reading Mode can also be included in Allevo main planning: Unlike Planning Mode, the reading has to be activated via a button in the Allevo toolbar. For this use case, the Allevo constant ACTIVE_FP must be set: The satellites for the Reading Mode must be listed there in "Value to" (as a sequence of 0/1). It is recommended to make this entry via the satellite assistant.

When Allevo Master is open, a button |Read Flexible Planning| appears, which can then be used for executing the reading function for the currently planned object.

5.3 Reading Activity Inputs

The Reading Mode requires additional constants: They define the planned postings that are to be read for direct activity allocation and order settlement (constants DA_TRANS_READ and OS_TRANS_READ). SAP procedures are listed for the respective constant; e.g. the operations

- RKP3 and RKP7 for direct activity allocation,
- RKPW and RKPX for secondary order settlement (automatically: KOAP).

Note: Enter the operations in the field "Value from". Multiple operations are to be separated by comma and without blank.

Pflege der FP-Festwerte

Import Export Prüfen Alle Festwerte hinzufügen Leere Festwerte entfernen

Datenauswahl

Kostenrechnungskreis 1000 Satellit 62 Schema READ 2 read order accounting

Parameter in Allevo FP	Wert von	Wert bis	Text
DA_TRANS_READ	RKP3		Buchungsvorgänge für Lesen DA
FD_ALLOC_TYPE	HIER_1_T		Feldname: Verrechnungsart
FD_KSTAR	INFOOBJ		Feldname: Kostenart /Konto
FD_OBJECT_TYPE	HIER_2_T		Feldname: Objektart
FD_QNTY_FIX	HIER_1		Feldname: Gesamtmenge fix
FD_QNTY_VAR	HIER_2		Feldname: Gesamtmenge variabel
FD_RCV_ACT_TYPE	PAROB_LSTA		Feldname: Empfänger-Leistungsart
FD_RECEIVER	PAROB_KOST		Feldname: Empfänger
FD_SENDER	KOSTL		Feldname: Sender / Partner
FD_SND_ACT_TYPE	CSKS_2		Feldname: Sender-Leistungsart
OS_TRANS_READ	RKPX		Buchungsvorgänge für Lesen OS

Figure 5.2: Example of reading activity inputs

In the Reading Mode, the sending cost center always writes into the field COBJECT of the satellite. Since this field plays a crucial role in the general Allevo environment as central selection criterion, it



may be useful to provide the sending cost center by a separate field (from Flexible Planning version 3.0.6, this is possible via the field FD_SENDER).

Usually, the activity allocations of the sending cost center are listed: These are usually displayed with a negative sign (from Flexible Planning Version 3.0.6, the cancellation required for this is possible by using the constant SIGN_MINUS).

Start Reading Mode

Set the object type KL on the start screen (stored as default). Due to the variety of possibilities in SAP, a selection based on the receiver object is not intended here.

Figure 5.3: Start of Flexible Planning in Reading Mode

An execution scheme *READ* must be created and started to use the Reading Mode. The planned quantities and receivers of the activities will then be read depending on cost element and activity type. If the receiver cannot be clearly identified due to program settings, the object type will be issued (e.g. OR for orders). Data are written in the satellite specified before.

Technical background: When reading, FP searches the COSS table for a sender either in the partner object (for activity allocation) or in the source object (for order settlement). The selected operations, the debit indicator S and the value type 1 for planning considered here. The respective receiver of the activity is then in the field "Object" of table COSS.

5.4 Reading Primary Costs

From Allevo version 3.5.9, FP Read also allows reading primary costs (i.e. table COSP) in order to derive further plan data from them if possible. This is controlled via the constant PC_TRANS_READ (in version 3.5 not contained in F4 selection):

- The entry in column "Value from" controls the operations to be considered (multiple entries as a comma-separated list).
- By default, data is read from the SAP table COSP; optionally, the table /KERN/ACOSP can also be entered in "Value to" (e.g. for reading data entered via Allevo objects).



Mapping and execution of the program are described equivalently to reading activity inputs (see notes above).

5.5 Reading With Individual Value Types

In the standard case, FP Read only selects data of value type 01 (= plan). In special cases it may be useful to also consider other value types.

Use case: Balance sheet of relevant values for orders or projects (e.g. investments, liabilities, receivables etc.) can be posted via "Statistical cost elements" (cost element type 90). Corresponding actual data can be found via value type 11 = statistical actual, plan data via value type 10 = statistical plan.

To read such data, the constant PC_VALUE_TYPE must be set (from 4.0 also available as OS_VALUE_TYPE, DA_VALUE_TYPE, KF_VALUE_TYPE and AT_VALUE_TYPE).

5.6 Reading the Data on a Monthly or Annual Basis

As in the Planning Mode, the constant MONTH_ACTIVE defines whether read plan data are stored monthly or as annual value. Correspondingly, the constants for quantity/value fields must be defined here as well:

- FD_QNTY_FIX and possibly FD_QNTY_VAR in annual planning and
- FD_QNTY_MNTH_FIX and possibly FD_QNTY_MNTH_VAR in active monthly planning.

In the last case, the names of the periodic quantity/value fields are to be indicated in the satellite without the last two digits (e.g. ZZIPP_MONTH, if the fields are named ZZIPP_MONTH01, ZZIPP_MONTH02 ...).

5.7 Reading Data in Subsequent Years

From version 3.3., this feature of /ALLEVO/FPR is only available to a limited extent. Reading can be performed via ProCED if required.

5.8 Relevant Constants

Reading Mode uses the same constants for mapping to satellite fields as Planning Mode. A few constants are added to control the reading individually.

- PC_TRANS_READ to activate reading of primary costs
- PC_VALUE_TYPE to define a value type while reading (from 4.0 also OS_VALUE_TYPE etc.)



6 Suggestions for Field Definitions in the Satellite

The following suggestions must be met when defining fields of the append in order to ensure a correct processing of data from the satellites. Further comments:

- Allevo partly offers own data types, especially for quantity and value satellite fields. These compensate for example for data types CURR and QUA and can be without reference currency or quantity units (which are usually of no importance when working with satellites anyway).
- Alternatively, fields in the satellite append are often created as “Integrated Type“ in SE11: This variant offers the advantage to enter an individual description for each field, improving also the understanding of the table content.

Field content	Corresponding constant parameter	Data element (component type)	Or direct type (with length)
Allocation type	FD_ALLOC_TYPE	/KERN/IPPALLOC	CHAR (2)
Receiving object type	FD_OBJECT_TYPE	/KERN/IPPPF_OBART (statt J_OBART)	CHAR (2)
Receiving object	FD_RECEIVER	/KERN/IPPCOBJECT	CHAR (24)
Receiving activity type	FD_RCV_ACT_TYPE	LSTAR	CHAR (6)
Resource	FD_RCV_RESOURCE	CO_RESSOURCE	CHAR (10)
Sending cost center	FD_SENDER	KOSTL	CHAR (10)
Sending activity type	FD_SND_ACT_TYPE	LSTAR	CHAR (6)
Cost element	FD_KSTAR	KSTAR	CHAR (10)
Statistical key figure	FD_STAGR	STAGR	CHAR (6)
Switching Scheme	FD_SWITCH_LAYOUT	CO_SWSMH	CHAR (3)
Currency	FD_CURRENCY	TWAER_ISO	CHAR (3)
TimeSet	FD_COLDEF	/KERN/IPPCOLKEY	CHAR (6)
Plan controlling area	FD_PLANKOKRS	KOKRS	CHAR (4)
Plan version	FD_PLANVERSN	VERSN	CHAR (3)
Plan year	D_PLANYEAR	GJAHR	NUMC (4)
Plan period from/to	FD_PLANPERIODFROM/TO	PERBL	NUMC (3)
Distribution key, fixed and variable	FD_DIST_KEY_FIX FD_DIST_KEY_VAR FD_DK_PRICE_FIX FD_DK_PRICE_VAR	DK_WKF	CHAR (4)



Field content	Corresponding constant parameter	Data element (component type)	Or direct type (with length)
Quantity fields for year and periods, fixed and variable	FD_QNTY_FIX FD_QNTY_VAR FD_QNTY_MNTH_FIX FD_QNTY_MNTH_VAR	/KERN/IPP_SAT_QUANT	DEC 15,3 (min) (instead of CURR)
Value fields for year and periods, fixed and variable	FD_VALUE_FIX FD_VALUE_VAR FD_VALUE_MNTH_FIX FD_VALUE_MNTH_VAR	/KERN/IPP_SAT_VALUE	DEC 15,3 (min) (instead of QUAN)
Activity price	FD_PRICE_FIX FD_PRICE_VAR	/KERN/IPP_SAT_PRICE (instead of TKGXXX)	DEC 11,2 (instead of CURR)
Activity price unit	FD_PRICE_UNIT	TKEXXX	NUMC (5)
Equivalence number	FD_EQUI_NO	AEQXX	DEC (3, 0)
Company code	FD_RCV_CCODE	BUKRS	CHAR (4)
Partner company code	FD_SND_CCODE	BUKRS	CHAR (4)
Functional area	FD_FUNCTIONAL_AREA	FKBER	CHAR (16)
Trading partner	FD_TRADING_PARTNER	RASSC	CHAR (6)
Sign	FD_SIGN		CHAR (1)