Profitcenter

ALLEVO

User manual

Valid from Allevo 4.3

Planning of costs, revenues, inventories and statistical key figures on Profitcenters

Planning on Profitcenters is largely integrated into the architecture of Allevo. This means that the familiar features of Allevo are basically available. These include:

- Status management

- MultiPage and MultiObject mode

- Representative planning (1/n via Profitcenters and account groups)

- Satellite tables

- Line item display

- Reporting

- Control of the Excel planning form via Allocation and TimeSets

At this point, we assume that the reader is familiar with the basic features of these functions or can access them via the Allevo documentation. In particular, reference is made to the explanatory manuals:

- Allevo & SAP

- Allevo & Excel

In the following, only the functions that are specific to Profitcenter will be discussed.

Content

Table of contents

1 Allevo PC Planning - Overview

1.1 Allevo in the Profitcenter planning environment

Often the plan values of the Profitcenter are generated by a fully automated process, called "plan integration" in SAP. The Profitcenter forms a superordinate aggregation level. If the plan integration is active, it ensures that

- a Profitcenter is automatically or at least at the push of a button "co-planned" by the assigned objects (cost center, order, WBS element);
- a plan data transfer from COPA is also available.

Therefore, there are usually special reasons for planning directly at the higher aggregation level of the Profitcenter.

Allevo supports the SAP planning functions in the Classic General Ledger as well as in the New General Ledger (represented by the summary record tables GLPCT and FAGLFLEXT). The requirements for planning on Profitcenters are described in the respective SAP documentation. With reference to the plan integration mentioned above, you can control reading and planning individually.

Allevo supports the SAP functions in the Classic General Ledger and in the New General Ledger (represented by the summary record tables GLPCT and FAGLFLEXT). With reference to the SAP functions of the plan integration, the reading of reference data and the active planning of new values can be controlled individually.

The usual transactions for direct planning of CO objects are also available for Profitcenters, e.g. for MultiPage or MultiObject applications (e.g. transaction /ALLEVO/PC).

Note:	In earlier program versions, an additional Allevo-specific function was available for the mixed display of CO objects and data from PC (row definition P, also called "Profitcenter integration"). Due to the multi-object possibilities of Allevo, this special function is no longer needed today and is also no longer supported.

Alternatively, Profitcenter planning can also be carried out via Allevo FP; e.g. as secondary planning in an Allevo CO layout.

As an extension of Profitcenter planning, Allevo also offers functions for balance sheet reporting and balance sheet planning (both for classic and New General Ledger).

1.2 Differentiation from SAP overhead cost planning

On the Profitcenter side, accounts are used instead of cost elements. This is initially a semantic difference, since the master data maintenance is partly identical. However, the Profitcenter has its own groupings for accounts. However, these can be transferred from Cost Center Accounting or from the FI balance sheet structure. This means that you can work with balance sheet accounts in Profitcenter planning that are not available in cost center planning.

Only "real" postings are made to the Profitcenter. Statistical postings are unknown. There are no value types; only four record types distinguish the character of the record.

The Profitcenter can be assigned to more than one company code. When you start planning, however, you must always select exactly one company code. The planning perspective is therefore more a combination of Profitcenter and company code. In Cost Center Accounting, the company code has no significance for planning, but is taken into account "internally" by the corresponding assignment of the object master record.

In the case of the partner Profitcenter, however, there is no need to plan the Profitcenter on the basis of the company code if there is a sender-receiver relationship. In the standard SAP system, the partner company code is not even available as a field in the planning layout. Therefore, the partner company code is simply ignored in the data record. Instead, the partner company can be activated in the planning form and entered in the planning.

Note:	From a structural point of view, it is therefore advantageous to assign the Profitcenter to only one company code.

There is a significant difference between partner relationships and cost center accounting: While in Cost Center Accounting an offsetting entry is made for the partner whenever an allocation is made, this is not the case in Profitcenter Accounting. It is therefore up to the planner to make the offsetting entry with the opposite sign. Reconciliation routines, such as the plan reconciliation of activity planning in Cost Center Accounting (KPSI), do not exist.

In addition, the distinction between total cost and cost of sales accounting plays a role in Profitcenter accounting. Cost center accounting always (necessarily) uses the period accounting method. In Profitcenter Accounting, however, you have a choice: If you choose the cost-of-sales method, the functional area usually intervenes.

One last difference that to mention here is the currency issue. While the cost center knows the object currency (company code currency of the cost center) and the controlling area currency in addition to the transaction currency, the Profitcenter knows the company code currency and the Profitcenter currency (local currency). The latter may or may not be the same as the controlling area currency.

1.3   PC planning in the S/4Hana environment

Simple Finance, the predecessor of S/4Hana, introduced a central table ACDOCA was introduced for actual line items. Totals record tables are no longer available. For plan data, the table ACDOCP was introduced later. Gradually, SAP functions for Profitcenter planning will also be converted to these tables. Allevo supports these new tables/functions as of Allevo version 4.0.

Allevo continues to support the previous SAP functionalities in the Profitcenter area, as long as the corresponding basic functions are activated in the SAP system. This ensures a smooth transition to the new world:

- The Classic General Ledger with the associated transactions is no longer available in S/4Hana. A reactivation is certainly only useful in exceptional cases and could possibly be done as a modification via SAP Note 2345118.
- Similarly, New General Ledger functions based on the summary record table FAGLFLEXT are only available with restrictions. In the first releases of S/4Hana, related transactions such as GP12N (or related interfaces) were deactivated or removed from the SAP Easy Access menu (see also SAP Note 2474069 - Reactivating GL Planning).

If the basic requirements are met, Allevo functions are still available, particularly for entering plan data: customers can therefore continue to work with the previous form of their profit center planning. As in the CO area, the Universal Journal is also optionally supplied with plan data (i.e. table AC-DOCP again).

See also SAPnote "2425255 - "Profitcenter accounting in the Universal Journal in S/4HANA On-Premise" with many general notes, which are also regularly updated by SAP.

2 Implementation in Allevo

2.1 Access to the Profitcenter planning

The usual transactions for direct planning of CO objects are also available for Profitcenters (e.g. /AL-LEVO/PC): planning can be done in MultiPage or MultiObject mode, for example.

Planning is done for the company code: This is a key difference to the CO planning functions of Allevo (where only the controlling area has to be specified as an organizational characteristic). The definition is done via the menu "Allevo >> Set company code". At the start, the user is asked to select a suitable company code if it has not been set before.

In current Allevo versions, the company code can also be displayed or be selected directly in the initial screen (see constant PC_REP_COMP_SEL).
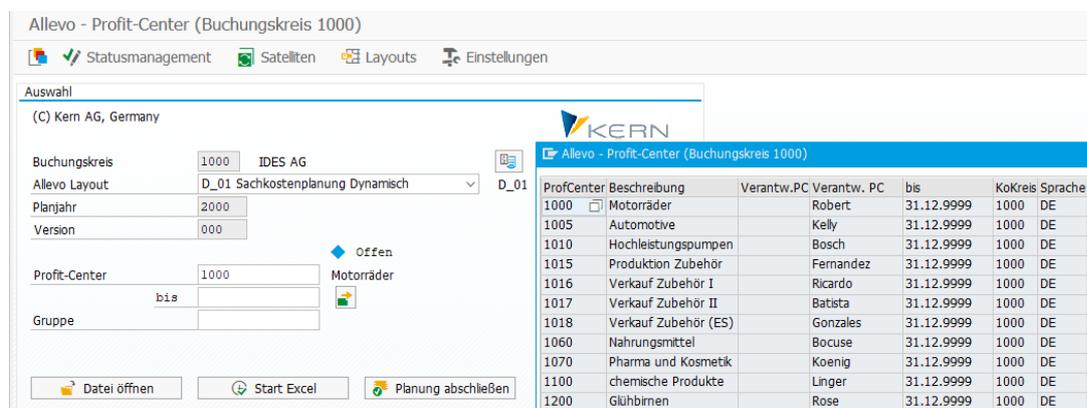
Figure 1: Profitcenter planning initial screen with F4 selection

The F4 selection for the Profitcenter displays a list of those Profitcenters that are assigned to the current company code. Constants can be used to control whether the reference data should be read for all company codes or only for the current one.

Alternatively, objects can also be selected via the Allevo Tree, where a Profitcenter group is usually specified on entry and the corresponding hierarchy is displayed. Objects in this hierarchy can be assigned to different company codes; if necessary, different company codes are also accepted with an automatic change of company code before the data is transferred to Excel.

2.2 Allevo Profitcenter Integration

Earlier Allevo versions contained an Allevo-specific profit center integration, which made it possible to read data from the corresponding Profitcenter while planning a CO object (e.g. a cost center) or to post directly to the profit center (e.g. revenues). This means that the Profitcenter was not called directly, but a substitute CO object (previously controlled via row definition P).

This function of Profitcenter integration is **no longer supported** as of Allevo 4.0. As an alternative, the extensive Allevo functions for direct PC planning can now be used, especially with a mix of different object types (MultiObject mode is now standard).

2.3   Special features about Allevo settings

Profitcenter planning has some special features that have to be considered in distinction to the Allevo settings of the overhead cost accounting.

2.3.1   Typical parameters of PC calculation

The following table first lists typical parameters in the classic or New General Ledger.

Parameter	Class.GL	New GL	Note
Ledger	8A	0L	See also constant PC_INT and LEDGER
Version Actual	0	1	See also constant PC_INT_VERSION
Summary record table	GLPCT	FAGLFLEXT (account) FAGLSKF (key figure)	PC Planning
Line items	GLPCA (Actual) GLPCP (plan)	FAGLFLEXA (actual) FAGLFLEXP (plan) FAGLSKF_PN (key figure)	
SAP plan transaction	7KE1 7KE3 7KE5	GP12N (account) FAGLSKF1 (key figure)	

2.3.2   Value type / record type

In Profitcenter Accounting, the record type is used (instead of the value type as in Overhead Cost Planning). The following options are available for selection:

> 0 = Actual

> 1 = Plan

> 2 = Allocation/dis Distribution Actual and

> 3 = Allocation /Distribution Plan.

This determination can be done as usual in the AllocationSet and/or TimeSet:
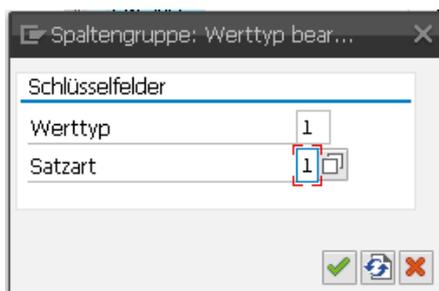
Figure 2: Initial screen for value type and record type

Since Allevo allows a combination of overhead and PC planning via MultiObject layouts, both specifications are proposed for input in parallel (with record type as a mandatory field in PC planning).

2.3.3 Operations

In some cases, it may be useful to read the values separately by transactions, e.g. in order to display the contents of direct PC postings in different rows from those of further allocation (in MultiObject mode, also with separate data for Profitcenters and participating cost centers).

Use case: In the standard case Allevo reads all values for the Profitcenter (e.g. direct costs and those from further allocation). If, in MultiObject mode, the participating cost centers are managed separately anyway, it may make sense to list only the direct costs in the PC row (i.e. to exclude all other transactions).

2.3.4 Currency type

By default Allevo recognizes the currency types T, O and C: different designations apply to them in comparison with the overhead cost planning, but also in comparison between the classic and the New General Ledger.

ID	New General Ledger	Classic General Ledger	CO Planning (for comparison)
T	Transaction currency (TSL)	Transaction currency (TSL)	Transaction currency (WTG)
O	Local currency (HSL)	Company code currency / local currency (HSL)	Object currency (WOG)
C	Group currency (KSL)	PC Local currency / Group currency (KSL)	Controlling area currency (WKG)
G	Fourth currency (OSL)	Used like 'C	Used like 'C

The correpsonding field groups are given in parentheses, as they appear in the tables GLPCT or FAGLFLEXT.

Currency type G is only available when reading data for the New General Ledger. In this case, Allevo reads via the field group OSL of FAGLFLEXT, whose contents are cleansed of currency fluctuations. Please note:

- The entry is stored in the F4 value help for TimeSet if the New General Ledger is set active via the PC_READTABLE constant.
- To enable the application in MultiPage mode, the constant is evaluated for all object types, not just for entry via PC
- When reading data without a New General Ledger, Allevo automatically switches to type C (i.e. to the controlling area currency in CO, for example).

2.3.5 TimeSets

Compared to overhead cost planning, the following settings of Allevo TimeSets have no meaning for planning in Profitcenter Accounting:

- Second currency/fixed share

■ Rate

■ Summarization and Allevo's own tables (not supported)

■ Delta Planning

Differences to value type / record type / currency type see notes section above.

Actual data for the New General Ledger are usually stored in version "1", so this version must therefore also be stored in the TimeSet for reading actual data.

As in the CO area, the periods 01 to 16 are supported in Allevo TimeSets; if a "period" 00 is entered, Allevo reads an annual carryforward (corresponds to behavior as in CO, but is mentioned here again for the sake of completeness).

2.3.6   Allevo AllocationSet / FP Allocation Type (Planning)

In the detailed settings for the AllocationSet, the following differences exist between Profitcenter Accounting and Overhead Cost Planning:

■ Load/unload indicator

This field is not available because the data is stored in a different structure in Profitcenter Accounting than in Overhead Cost Accounting. In Profitcenter Accounting, for example, the correction of a debit plan is also posted as a credit if the new plan value is lower than the old one (and vice versa).

■ Value type/record type

See notes in the previous section.

Which AllocationSets can be used depends on the general ledger you are working with. You will find corresponding overviews in the following sections.

Since Allevo 3.3, Profitcenter planning is also fully supported by the add-on module FP. Here is the assignment between the Allevo row definition and the allocation type used in FP when used within Profitcenter planning:

AllocSet	FP AllocType	Meaning
AC (A,B)	AC	simple posting with account
C,D	PP	Posting to account & partner Profitcenter
	FA	Posting to account, partner Profitcenter & functional area
	PT	Posting to account, partner Profitcenter, functional area & company number
I,J	AA	Posting to a balance sheet account
KF (Q,R)	SH	Posting to statistical key figure (Classic General Ledger only)

There may be differences between the planning functions in the New and Classic General Ledger. Please contact our support for clarification.

3 Read reference data for Profitcenter

3.1 Function overview

The read functions are performed largely according to the systematics of the Allevo read functions. Technically, the system reads from the summary record tables GLPCT or FAGLFLEXT, instead of the tables COSP and COSS.

The constant PC_BEKNZ can be used to display debit and credit postings in separate rows in the Allevo master. This constant must be set additionally, as the entries in the AllocationSet alone are not sufficient for PC planning alone (background is a different understanding of the function in the PC and CO tables).

Profitcenter accounting has its own ledger (e.g. "8A" for Classic General Ledger, "0L" for New General Ledger); deviations from this must be stored in the constant LEDGER (see F1 documentation for the constant).

3.2 Relevant AllocationSets

The following table shows the functions of the Allevo AllocationSets when reading reference data. The different results are mainly due to the additional parameters that can be used during the selection. Certain column keys have to be filled in to the transfer of these parameters from Excel (see information in brackets).

AllocSet.	Reading costs and revenues	
AC	Values and quantities for costs/revenues	
KF	Statistical key figures (total values, constants, constants as average)	
TX	Text row	

Additional constants can be used to customize the behavior when reading reference data (see the following sections).

3.3 Reading via additional features (e.g. functional area)

Allevo allows you to narrow down the read data using additional features that can be specified very individually in Excel (e.g. narrowing down by using an additional functional area). Specific pointers are provided for the transfer from Excel to SAP.

Review of previous Allevo versions

Allevo used column keys with special meaning (currently these are CY_KEYR and CY_KEYSA). The constants READ_WITH_KEYR and READ_WITH_KEYSA control which additional characteristics are to be taken into account and for which AllocationSets they are to apply. The following characteristics can currently be used as additional characteristics:

- Characteristics such as functional area, partner functional area and cost center group are handled specifically (restriction to functional area can also be used for planning in NGL).
- Optionally, any field of the relevant database table can be addressed.

For details, see the F1 documentation on the READ_WITH_KEYR constant. These additional restrictions also apply to Allevo applications in the CO area (e.g. in cost center planning).

3.4   Empty field contents as selection parameters

In some cases it is necessary to limit the selection of data so that only data records with empty field contents are considered. This function can be especially helpful when reading data for Profitcenter Accounting.

For example: Only direct postings to Profitcenters are to be considered; but not data records that have been created via postings through from cost centers. It is therefore necessary to query for an empty origin object (field HOART).

The function is activated using the READ_EXCLUDING_FIELD constant.

3.5   Relevant company code when reading reference data

By default, the system reads reference data for the current Profitcenter and for the current company code (although reading via a defined company code makes more sense for performance reasons). The systems determines the relevant company code as follows.

- When calling the direct Profitcenter planning (e.g. Ta /ALLEVO/PC), Allevo reads the reference data using the company code entered when entering the respective Allevo transaction.
- Optionally, the company code can be selected individually in the initial screen of the respective transaction (see constant PC_REP_COMP_SEL). The F4 selection help displays all company codes for the current controlling area.
- In reporting mode (i.e. /ALLEVO/PCREP), you can also enter * to read a total for all company codes.

Reading PC data when accessing via a CO object type

When accessing via a CO object type (e.g. with MultiObject or MOWO use case), the company code is not explicitly queried in the Allevo initial screen. However, if Profitcenter data is to be read in the corresponding master, it is usually necessary to specify the company code. Possible option:

- In the standard case, Allevo takes the specification from the header data provided by the Excel side (see the "SheetObject" area in the Allevo Master, or CC_BUKRS in earlier versions). Often this specification is taken from the master data of the respective entry object, e.g. an Allevo object that is to represent a company code. If a * is transferred from the Excel side, then Allevo reads over all company codes.
- Optionally, the restriction can also be made on the Excel side by specifying the database field BUKRS in the dialog corner (see header row READ, in earlier Allevo versions it was defined by the constant READ_WITH_KEYR). The advantage of this is a row-by-row display on the Excel side; e.g.

with a Profitcenter repeated over several rows, where a different company code applies for each row. In this case the entry in the header area (or CC_BUKRS) must contain a * again.

The functions described here also apply to planning.

Reading without company code restriction

In the following cases, data is read without restriction to a company code:

- When the NO_COMPANYCODE constant is active (see documentation for the constant).
- Extended reporting function controlled by the constant PC_REP_COMP_SEL with entry * for the company code in the initial screen of the respective transaction (see also the general description in section 5.12 and the application in section 6 on data selection for the balance sheet report).
- Start via the CO object type and entry * in the "SheetObject" area (or for CC_BUKRS in earlier versions).

Reading PC data separately via individual Excel sheets per company code

For Profitcenters or PC groups that are assigned to several company codes, the breakdown by company code is carried out in this display via several Excel sheets (MultiObject application via the so-called FISH variant of the Allevo master).

This display variant can also be used for planning and is described in more detail in section 5.9.

3.6 Elimination of intercompany sales

Using the IBV_ELIM constant, Allevo also supports the elimination of intercompany sales in the Profitcenter planning environment. For example, if a cost center provides an activity allocation (or other secondary posting) to an order in the same Profitcenter, this relationship is not shown when reading from the Profitcenter.

This function applies both when reading data for the Classic General Ledger and when using the New General Ledger (see constant PC_READTABLE).

For more details, see the documentation for the IBV_ELIM constant.

4 Profitcenter planning Classic General Ledger

4.1 Overview

For Profitcenter planning, the Classic General Ledger uses the summary record table GLPCT as standard; plan data is entered using transactions such as 7KE1. For balance sheet planning, table GLT0 is used and GP12N is the transaction for entering plan data. "8A" is the standard ledger; actual data is usually stored in version "0" (as in CO).

Note:	In S/4Hana, the Classic General Ledger with the corresponding transactions is no longer available. Reactivating it should only be done in exceptional cases and could possibly be done as a modification via SAP Note 2345118.
	See also SAPnote "2425255 - Profitcenter Accounting in the Universal Journal in S/4HANA On-Premise" with many general notes, which are also regularly updated by SAP.

4.2 Reading reference data

When reading reference data, the PC_READTABLE constant can be used to select a different table (e.g. a customer-specific table). However, it must then be ensured that all the fields Allevo needs for data selection are available. This constant also activates reading via New General Ledger (see section below).

By default, the ledger "8A" is read; different settings must to be stored in the "Value to" column of the LEDGER constant (see the F1 documentation for the constant).

The relevant AllocationSets are described in the section 3 section above.

4.3 Profitcenter planning (AllocationSets)

Since SAP has not provided a BAPI for Profitcenter planning for the Classic General Ledger, Allevo uses an automated batch input for the usual planning transactions (i.e. 7KE1 for costs/revenues, 7KE3 for inventories and 7KE5 for statistical key figures).

For reasons of data security, the transfer of data records is strictly sequential. The interface is therefore less efficient than solutions using BAPI functions. For large volumes of data, it can be helpful to use an accounting via Allevo-FP and the background processing that is possible there (see Techn. prerequisites in the next section).

Data is always posted to ledger 8A: this results from the batch input procedure used and cannot be controlled externally.

AllocationSets

Here are the AllocationSets that can be used, specifying the planned transactions for the Classic General Ledger.

Z.Def.	Content	Transaction

AC	Plan costs/revenues with or without quantities	7KE1
AC	Plan stock accounts (account type is recognized independently by Allevo)	7KE3
KF	Statistical key figure constants or summed values	7KE5
TX	Comment	

The reference data is read taking into account the balance carryforward.

Functions for statistical key figures are currently only available for the Classic General Ledger.

Special feature for planning with quantities

When planning quantities to Profitcenters using transaction 7KE1, a unit of measure must always be specified, especially for the call via batch input used by Allevo. Allevo knows two methods to determine the correct unit of measure:

- Transfer from the master record of the corresponding cost elements (table CSKB).
- Transfer from the Allevo master using the column key CY_KEYRA (otherwise used for the receiver activity type). This entry overwrites the value from the cost element master record if necessary.

Rounding: From the Allevo master, quantities are always transferred to SAP with 3 decimal places. However, since many SAP units of measure do not allow decimal places (e.g. if they are unit-based), they are rounded before posting according to table T006 (based on the current totals, e.g. over all periods).

Note:	Annual planning with distribution to months: As of Allevo 3.4, it is possible to use distribution keys when posting in the Classic General Ledger; the entry for "Fix" is used (suffix "_DKF" to the column key, transfer as usual via batch input to the SAP transactions).

4.4 Additional features during planning

As of version 3.4, Allevo offers functions to take other characteristics into account when reading and writing (e.g. characteristic functional area). If necessary, the need for this must be clarified in the implementation project. See also the notes on technical requirements in the next section.

If AC is to be planned for a functional area in the AllocationSet (set via constant READ_WITH_KEYR with entry "FUNC_AREA"), then Allevo automatically uses layout 8A-115 of SAP transaction 7KE1 during posting.

Planning with functional area is also realized for FP.

Posting with quantities is not possible via layout 8A-115: Allevo uses layout 8A-113 or 8A-114 for this (but the functional area is not considered there). For posting with partner Profitcenter via 8A-114 the partner must be entered in the Excel column (TimeSet CY_KEYS, CM_KEYS).

4.5 Company code row by row

In the standard case (see above), Allevo posts with the company code that is available in CC_BUKRS: consequently, only the same company code can be planned for each Excel sheet. If different company codes are to be addressed in the same master, the MultiPage mode can be used.

As of Allevo 3.5, the company code can also be specified for each row in Excel, e.g. via an entry for the CY_KEYR column key. Two conditions must be fulfilled for this:

- The relevant database field BUKRS must be entered as a parameter in the AllocationSet (in the example here at "Use KEYR field as").
- A * must be entered at CC_BUKRS.

The function described for displaying the "Company code row by row" applies to normal PC planning in GLPCT and the balance sheet report via GLT0. It is available in planning from Allevo 3.5; in version 3.4, only a read function via equivalent constants was implemented (for details, see constant READ_WITH_KEYR or notes in section 6).

4.6 Technical requirements

Allevo uses a batch input procedure for all postings in the Classic General Ledger with a call of the relevant SAP planning transactions (for balance sheet planning, see notes in the previous section).

For Profitcenter planning, the following layouts has to exist and be entered in the planner profile:

- Transaction 7KE1 for cost/revenue planning with the following layouts:

    o 8A-113   PC, account + quantity

    o 8A-114   PC, account, Partner PC + Quantity

    o 8A-115   PC, account, functional area without quantities (as of Allevo 3.4)

- Transaction 7KE3 for stock planning with the following layout

    o 8A-413   PC, account + quantity (stocks 7KE3)

- Transaction 7KE5 for planning statistical key figures with Layout

    o 8A-311   PC + stat. key figure

If you want to access layout 8A-115 for postings via the functional area, you can use READ_WITH_KEYR, for example, to make the relevant functional area available via Excel. In principle, other parameters can also be transferred; if necessary the application must be clarified with our support if required (SAP layout must be available for this).

Customized layouts

As of version 3.4, Allevo supports the use of additional SAP custom layouts, as described above for the functional area.

In special cases, the use of additional characteristics may be desired, which are not supported by the SAP standard layouts (in particular transaction 7KE1, e.g. when executed via FP Allocation type FA or PT). These additional features can only be used if suitable customer-specific 7KE1 layouts are available.

Note:	The use of customer-specific layouts should no longer be necessary as of 3.4. The requirements are described here only for compatibility with earlier versions. The need is to be clarified in the implementation project or together with our support.

Currently, the following naming conventions must be observed for the SAP layout:

- Z8A-113 for PC planning using a combination of account, partner PC and functional area
- Z8A-114 for PC planning using a combination account, partner PC, functional area and partner company

These layouts must also be stored in the profile of the user; background processing is not possible for customer-specific layouts when called via FP.

4.7 Permissions

The following authorization checks must be carried out before you can call up the above transactions and layouts for Profitcenter planning:

- Planning authorizations for the Profitcenter are checked using the authorization object *K_PCA* with *co_action 1002*, *1003* and *3027* (see SAP function module K_PCA_RESP_AUTHORITY_CHECK).
- Due to the batch input technology used, the following two general authorization objects must also be stored with the user:

    o *K_TKA50* (activity 16, authorization group *)

    o *G_GLTP* (activity 02, 03, ledger 8A, record type *, version *)

    The * indicator for the authorization group or versions can of course be replaced by concrete entries. This should not make much sense for the record type.

- Authorization object *G_GLTP* is also required in particular for planning at company code level (balance sheet planning).

4.8 Performance

The batch input procedure on the mentioned SAP planning transactions is less performant compared to calling BAPIs as it is common with other Allevo object types.

Of course, this is especially noticeable when many positions are posted at the same time. The following measures can be helpful:

- Use of constant NO_ZERO_DELTA (possible from Allevo 3.4).
- If necessary, also use constant PLAN_PC_IN_LOOP for a blockwise transfer of PC plan data to the batch input (only for monthly planning).

- Posting via Allevo FP with execution in the background (only possible with SAP layouts)

5 Profitcener Planning New General Ledger (NGL)

5.1 Overview

The SAP table from which Allevo reads the relevant reference data is set via the PC_READTABLE constant (usually this is FAGLFLEXT, but it can also be a customer-specific table). The conversion to the New General Ledger is therefore very simple for reading.

The situation becomes more complex if planning for the New General Ledger is also to be done directly via Profitcenters in Allevo and no longer derived from other objects with SAP plan integration (e.g. via cost centers or orders). In this case, Allevo uses the BAPI_FAGL_PLANNING_POST function module, which can map different scenarios. As an initial overview:

- The BAPI needs information about which characteristics to use when posting. There are mandatory parameters and optional parameters. For example, a posting can be made with or without a functional area or segment; if required, completely without a Profitcenter.
- Some characteristics of the posting are automatically derived by Allevo (e.g. company code from master record to Profitcenter). Other characteristics can be entered individually in Excel: the transfer to SAP then takes place with the help of special column keys and controlled via constants.
- The BAPI compares new plan data with previous values and automatically generates a delta posting (zero data records should therefore not occur). For this BAPI-internal analysis the previously valid data must be read: care must be taken to use the same characteristics as for writing (otherwise the plan values will multiply).
- For individual characteristics, the BAPI runs through its own determination to determine default values if a characteristic is not provided via the interface. It can therefore also occur that a characteristic is unintentionally filled in SAP (i.e. without default from Excel).
- In the case of monthly values in particular, new plan values and read plan values are often in the same cell. In this case in particular, it is important that the Allevo read functions apply exactly the same characteristics as they do when plan data is transferred (example: reading without a functional area, but writing with a functional area will certainly lead to inconsistencies).

Statistical key figures: as of Allevo 3.5, statistical key figures can be read for the NGL (previously, data was read from table GLPCT, which still belongs to the Classic General Ledger). Planning statistical key figures for the New General Ledger is only supported from Allevo 4.0. Alternatively, in individual cases it can also be useful to store planning data via CO objects in SAP, e.g. via suitable cost centers.

Useful SAP transactions for the NGL:

- GP12N   Entry of plan data (years and periods)
- FAGLSKF   Post actual stat. key figures (alternatively KB31N via cost center)
- FAGLSKF1   Post plan stat. key figures (alternatively KP46 via cost center)

Calling transaction GP12N can also be useful for test purposes: to access planning for the New General Ledger, select planner profile SAPFAGL with totals table FAGLFLEXT (standard case) when calling the transaction.

Notice:	In S/4Hana, the New General Ledger based on summary record table FAGLFLEXT is only available with restrictions. In the first releases of S/4Hana, related transactions like GP12N (or the above mentioned BAPI) were deactivated or removed from the SAP Easy Access Menu (see also SAP Note 2474069 - Reactivating GL Planning).

5.2 Read reference data

The reading of reference data from the New General Ledger tables is activated via the PC_READTABLE constant (entry FAGLFLEXT in column "Value from"). After that, data is read from the summary record table of the same name instead of GLPCT as in the Classic General Ledger.

If the Allevo plan functions are also active for the New General Ledger (see next section), Allevo automatically reads from the table FAGLFLEXT (even without an entry for PC_READTABLE).

Actual data for the New General Ledger (record types 0 and 2) are kept in version "1" by default; consequently, this version must also be stored in the TimeSets for reading actual data (and not "0", as it is usually the case in CO transactions.). Different versions must be entered at constant PC_INT_VERSION, if necessary.

The relevant AllocationSets are described in the section 3 section above.

Central Tables to the New General Ledger:

- FAGLFLEXT   Summary record table
- FAGLFLEXP   Plan line items (evaluation via SAP transaction FAGLP03)
- FAGLFLEXA   Actual line items (evaluation via SAP transaction FAGLL03)
- FAGLSKF   Stat. key figures: Totals records
- FAGLSKF_PN   Stat. key figures: Line items (actual and plan)

By default, the ledger is read with "0L"; deviating settings are to be stored in column "Value to" at constant LEDGER (see F1 documentation for constant).

Notice:	Equivalent reports for the New General Ledger are provided, for example, by SAP transaction FGI3.

Special Notes:

- *Display of line items for New General Ledger*

  The display of line items in Allevo is available as of version 3.3.36 (see also documentation on the constant BUTTON_LINEITEM). In the case of the New General Ledger, Allevo always jumps to the FI line items according to transaction FS00; no jump is possible for plan data.

- *Statistical key figures*

  Reading statistical key figures for the New General Ledger is available as of version 3.5 (table FAGLSKF). Previously, GLPCT was accessed.

  Planning of key figures in the NGL is supported from Allevo 4.0.

- *Currency type for reading OSL fields*

The additional currency type G is available when data is to be read for the New General Ledger: this allows values to be read that have been adjusted for currency fluctuations. For more details see section 2.3.4.

5.3 Basis for planning via Allevo

The following are the basic steps that must be considered for planning into the New General Ledger; the main topics:

- Activation via Allevo constant PC_NGLA_PLANNING
- Definition of the optional characteristics to be taken into account for posting
- Allevo Master and definition for call (e.g. entry object).

Activation of the planning via constant

The posting to the New General Ledger is activated via the Allevo constant PC_NGLA_PLANNING: the constant is also taken into account by FP, as long as FP is connected to this layout. In this case, posting to SAP takes place via the function module BAPI_FAGL_PLANNING_POST = transfer of planning data to the New General Ledger.

Notice:	The module is only available from EHP3 of SAP ERP. For versions prior to this, Allevo offers a solution with limited functionality via transaction GP12N (see section 5.6.3).
	Planning statistical key figures for the New General Ledger is only possible from Allevo 4.0.

The technical requirements can be found in the section 5.7 further below. Except for minor deviations the mentioned BAPI offers the same possibilities as the equivalent SAP planning transactions, so e.g. GP12N (a restriction applies e.g. to the currency used).

A few basic Allevo settings are required for posting at Profitcenter level. Currently the AllocationSet AC is used. Some basic properties:

- Annual/Monthly Planning:

  The SAP BAPI can only post monthly values. If yearly values are transferred from the Excel side, Allevo automatically divides the values linear among the months.

  Planning via annual value with SAP distribution key is currently not yet possible.

  See also section 7.3 with note on balance sheet accounts in balance sheet planning.

- Ledger:

  By default, planning is done with ledger "0L", unless another value is entered in "Value to" at constant LEDGER (see F1 documentation for the constant). If there are several entries there, the first ledger in the list applies for the plan data entry.

- Currencies:

In contrast to the transaction GP12N the BAPI always expects the transferred plan data in the company code currency. The translation into controlling area currency or parallel local currencies is done automatically in the BAPI with the exchange rate type defined in Customizing (normally exchange rate type M, see "Technical Requirements" below). The BAPI determines the date for the exchange rate internally equivalent to the behavior in GP12N.

If plan data is to be entered in a different currency, it may be necessary to perform a conversion on the Excel side beforehand: the Allevo cell definition CC_CURRATE can be helpful for this (see Allevo Excel manual).

The currency key in the Allevo TimeSets has no significance for planning in the New General Ledger, but only for reading reference data.

Special cases of planning can be mapped using an additional control table, which is described below (e.g. planning at company code or segment level according to segment reporting under US GAAP and IFRS).

5.4   Determine characteristics of a posting via BAPI

The BAPI allows you to enter planning data with a wide variety of characteristics. As a result, several factors must be considered when setting up the associated Allevo functions.

The following table shows all characteristics that must be available as **mandatory fields** when transferring plan data to the BAPI and which are also always filled by Allevo accordingly (field name according to FAGLFLEXT).

Feature (field)	Meaning	Source
RRCTY	Record type	fix 1 = plan
RVERS	Version	Allevo plan version
RLDNR	Ledger	See Ledger constant (0L as default)
RBUKRS	Company code	Allevo company code
RYEAR	Business year	Allevo plan year
RACCT	Account number	Allevo account
RUNIT	Base unit of measure	Technical specification: determined internally
RPMAX	Maximum period	Technical specification: determined internally

Figure 3: Mandatory fields for posting in the New General Ledger

You can see that the Profitcenter is not a mandatory field, which gives the possibility to post independently from the Profitcenter.

All the more important are the **optional characteristics** that can be specified for each data row. The following list contains the associated database fields and also (in brackets) the abbreviations as used in the BAPI interface. The latter are also important for configuration in Allevo (see notes below).

The "Source" column describes how a characteristic can be filled via Allevo / FP, or whether there is an automatic determination in the BAPI for this (which may not even be wanted).

Characteristic (field name)	Meaning	Source
KOKRS (CO_AREA.)	Controlling area	Current Allevo controlling area
PRCTR (PROFIT_CTR]	Profitcenter	Current object from Allevo Master or FP posting row (BAPI internal derivation would take place via the cost center).
SEGMENT (SEGMENT.)	Segment for reporting according to US GAAP / IFR	Allevo takes over the entry from the master record of current Profitcenter (BAPI internal derivation would be via Profitcenter or substitution).
RASSC (TRADE_ID]	Partner company number	Allevo column key CY_KEYSA; for FP via constant FD_TRADINGPARTNER
RFAREA (FUNC_AREA]	Functional area	Allevo column key CY_KEYR; for FP via the constant FD_FUNCTIONAL_AREA. (BAPI internal derivation is done via entry in revenue account, cost center or substitution; see SAP documentation for "Rules for deriving the functional area for postings", or SAP Note 740519).
PPRCTR (PART_PRCTR]	Partner Profitcenter	Allevo column key CY_KEYS; for FP via constant FD_SENDER (BAPI-internally derived from partner cost center)
PSEGMENT (PARTNER_SEGMENT)	Partner segment	Transfer from the master record in the partner Profitcenter

Figure 4: Optional fields for posting in the New General Ledger

Once again, it should be noted that the Profitcenter in particular is an optional parameter. In the standard case with planning on Profitcenter in Allevo, the parameter PROFIT_CTR should therefore also be active, together with the controlling area (details follow below).

These additional characteristics are taken into account in the following BAPI-internal functions:

- BAPI internal reading of existing plan data,
- Determine the difference between the read planned values and the target value from Allevo,
- Writing the determined delta values into the SAP database tables (e.g. FAGLFLEXT).

To ensure that these BAPI-internal functions lead to a correct result, all parameters (used in the interface) should have a valid value when passed so that this value can be used when reading and writing the data BAPI-internally.

This rule applies in particular to those parameters for which a separate determination is stored BAPI-internal, which is always run if a parameter is relevant but no value has been passed in the interface. Unfortunately, this determination is only run for the planned value. This poses the following danger:

- In the first step, BAPI is read internally with an empty parameter, e.g. with an empty function area.
- Existing plan values with set functional area are therefore not found.
- The BAPI calculates the "difference", so it completely reposts the plan value transferred via interface. The plan data belonging to the characteristic combination consequently increases with each transfer of plan data (which can be easily read in the corresponding SAP reports).

Allevo must therefore have unique contents for the relevant parameters (e.g. functional area). Possible ways to achieve this:

- Entry under the appropriate column key in the Allevo master (e.g. for CY_KEYR).
- Entry of a default value in Allevo's own additional table (see notes below): however, this can have the disadvantage that Allevo reference data is read with deviating characteristic values. This again results in the risk of skewed values, especially for monthly values with only one cell for new and old value.

Notice:	In principle, the planning BAPIs also allow the use of customer-specific Z* fields as characteristics. However, this function is currently not yet supported by Allevo.

Set optional planning characteristics

The Allevo table /KERN/IPP_PCNGL controls which of the above-mentioned optional planning characteristics are to be used: the table describes rules of an Allevo-specific determination logic for these characteristics. Entries are created via transaction /ALLEVO/PC_NGL_MNT. Here is an example:

Tabellenpflege:

📤 Upload   📥 Download   🔓 Prüfen

KKrs	BuKr.	Layout	Merkmal	Aktiv	Default Wert
1000	*	*	SEGMENT	X	
1000	*	WHF1	FUNC_AREA	X	
1000	1000	*	PART_PRCTR		
1000	1000	*	PROFIT_CTR		
1000	1000	ESR	PROFIT_CTR	X	

Figure 5: Definition of characteristics when posting to New General Ledger

The first three columns describe the area of validity of a determination rule: so controlling area, company code and/or layout. The object type is missing: in the default case, Allevo assumes that a layout is to be selected for the object type PC. If planning is carried out with a different object type (e.g. Allevo object), the object type can be changed within the F4 selection. Double-click on the header area in the pop-up to display the selection: After a new selection, the layouts of the other object type are

also offered in the popup (layouts with the same name may also be offered twice; before Allevo 4.0, a layout with the same name may also have had to be created for PC).

The next column describes the characteristic which is determined by the respective rule (according to the list in Figure 4). Explanations to the example above:

- Line 1 specifies that all postings in controlling area 1000 are always accompanied by an indication of the segment.
- Line 2 specifies that the functional area is to be transferred in addition to the layout WHF1 (the corresponding abbreviation for the functional area must then of course also be provided via upstream Allevo functions, see e.g. note in the next section).

In the standard case of Profitcenter planning, the table should at least contain entries for the characteristics CO_AREA and PROFIT_CTR.

An X must be entered in the penultimate "Active" column so that the associated row is taken into account when plan data is transferred to the BAPI. Without this switch, the characteristic is only taken into account when reading.

Im-portant:	Normally, the switch should be set to avoid deviations in BAPI-internal reading and writing, or in the delta calculation (the dangers have already been pointed out above).
	This is especially true for characteristics for which a BAPI-internal determination is made.
	On the other hand, this makes it possible to explicitly post a data record with an empty characteristic (so characteristic not set to active for planning), and also to read only the row with the empty characteristic value beforehand for delta determination (but must then also be transferred empty from Allevo).

The last column allows to additionally store a default value: it will be used automatically if a characteristic could not be set in the upstream Allevo process.

Additional rules of finding:

- Characteristics that are not entered in table /KERN/IPP_PCNGL are also not taken into account during posting. The associated content that was transferred by Allevo (or FP) is therefore ignored.
- If the /KERN/IPP_PCNGL table does not contain any entries for the current determination parameter (controlling area, company code, etc.), Allevo transfers a list with all seven optional names to the BAPI interface. The characteristics should then also be supplied completely via the interface (or they result from SAP's own internal determination). Normally, however, it makes much more sense to use the control functions of the table in order to have reproducible behavior.

Special case for field ZZHOART/HOART: via SAP Note 1555535 and others, the origin object type is subsequently also introduced for the New General Ledger. If this field exists and is relevant for posting, it must also be entered in table /KERN/IPP_PCNGL (otherwise an error message appears when posting, e.g. "Field ZZHOART missing in structure FIELDLIST").

Important:	Changing the characteristic combinations in table /KERN/IPP_PCNGL should be avoided at all costs during a planning phase and within a version (otherwise there is a risk of data statuses that are not traceable).

5.5 Profitcenter planning (standard case)

The Profitcenter is an optional parameter when posting via BAPI to the New General Ledger. So in this standard case of Allevo planning the parameter PROFIT_CTR should be active in the table /KERN/IPP_PCNGL; together with controlling area in CO_AREA (details see above).

5.6 Other use cases

The last section described the basic principles that must be taken into account for planning into the New General Ledger. In the following, individual cases of application are described in more detail.

5.6.1 Segment planning, etc.

As with balance sheet planning (see last section), control via table /KERN/IPP_PCNGL also allows planning via other characteristics, in particular planning on segments (= business line of a company for external reporting according to US GAAP and IFR). The basic functions are as already described for balance sheet planning.

The relevant segments must be created in Customizing for the New General Ledger; the segment is transferred again from the current (representative) Profitcenter for the Allevo posting.

Please coordinate these functions with our support if required.

5.6.2 Use additional features (e.g. functional area)

Section 3.3 describes how reference data can be read, e.g. restricted to a specific function area, whereby the respective function area is specified via an entry in Excel (see documentation on constant READ_WITH_KEYR or READ_WITH_KEYSA).

The entry for the functional area can also be used for planning with the NGL. Two conditions must be met for this:

- Transfer from Excel to SAP using the column keys CY_KEYR and CY_KEYSA as for reading (or via constant FD_FUNCTIONAL_AREA when called via FP).
- Entry in table /KERN/IPP_PCNGL, so that the functional area (so optional parameter of an NGL posting) is also transferred to the SAP-BAPI.

Under these conditions, the Allevo read and write functions also fit together. It is advisable to create several AllocationSets if rows with equivalent posting combinations but different functional areas are to be displayed within a master.

If the CY_KEYR or CY_KEYSA column keys contain multiple functional areas on the Excel side, Allevo uses the first entry when planning.

Planning with partner functional area is currently not possible.

5.6.3 Special case: PC planning for NGL via batch input for GP12N

The SAP BAPI used by Allevo was only introduced with EHP3. For older releases, planning is possible to a limited extent by execution via batch input to SAP transaction GP12N.

This variant of the plan data transfer is activated via the NEWGL_PC_CURRENCY constant: in the column "Value of" the abbreviation of the currency is to be entered, with which is to be booked. The constant will only be necessary in exceptional cases and is therefore not included in the F4 selection.

5.7 Technical requirements

In order to be able to plan with Allevo in the New General Ledger, the function module BAPI_FAGL_PLANNING_POST must be available, or BAPI_ACC_POST_STAT_KEYFIGURE for posting stat. key figures.

SAP delivered the enhancement package EHP3 with the business function FIN_GL_CI_1. This function must therefore at least be imported into the system (see SAP Notes 1070629 - FAQs: New General Ledger migration, as well as 826357 Profitcenter Accounting and New General Ledger).

Notice:	Allevo independently checks whether the required SAP planning BAPI is available in the customer system.
	As of version 3.4, the standard Allevo delivery already contains the basic functions required for planning via the New General Ledger (without additional transport). However, posting Stat. key figure is only possible from Allevo 4.0.

For planning in the New General Ledger, SAP offers its own customizing, which is independent of the customizing of classic Profitcenter Accounting:

SPRO: "Financial Accounting (New) >> general ledger Accounting (New) >> Planning".

Basically, the same prerequisites must be met for the above BAPI as for the equivalent SAP planning transaction GP12N. Some central points:

- Activation of relevant plan periods via transaction GCP5; an entry for account type G (= special ledger) is entered in the table T001B. Without suitable periods, the SAP error message GU501 (= period xx not opened for posting) appears during plan data transfer.
- Set up plan version for ledger 0L via transaction OKEQ (entries in table T894, otherwise SAP message GLPL710 appears).
- Activate plan year and plan version company code via transaction GLP2 (otherwise SAP message GLPL776 appears).
- Activation of manual planning via transaction GLPV; otherwise message GLPL719 appears if no entry is made (the transaction OBP6 is only relevant for the Classic General Ledger).
- A document type for plan line items must be active with assignment of matching number ranges.

In the standard case it should be document type P0 (= planning PC New GL) together with the number range 01. This master record can be created via the transaction SM30 with maintenance view V_FAGL_T889A (see also SAPnote 2518263).

For number range 01, an interval must be stored for each company code and plan year (input via the transaction FAGL_PL_LC).

- Depending on the installation, entries for number range object GL_FLEXPL or FAGL_PL_LC may also be required (SAP message GI269 may appear, control via transaction SNRO or GB02).

For the mentioned BAPI (and thus for planning via Allevo) additional settings may be required. In SAP Customizing for planning for New General Ledger, these settings can be found under the title "External plan data transfer":
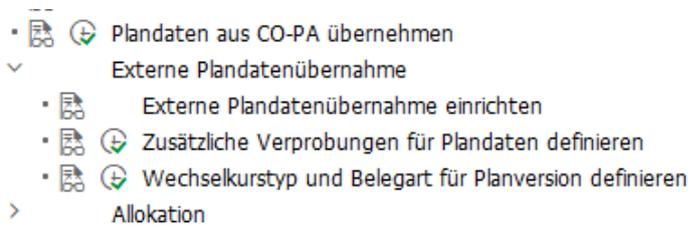

- Plandaten aus CO-PA übernehmen
  - Externe Plandatenübernahme
    - Externe Plandatenübernahme einrichten
    - Zusätzliche Verprobungen für Plandaten definieren
    - Wechselkurstyp und Belegart für Planversion definieren
  - Allokation

Figure 6: "External plan data transfer" menu item

Notice:	If the mentioned submenu "External plan data transfer" is not visible in SPRO, please refer to the SAP Note 2184576.

The most important entries here are:

- The "Additional validation ..." menu item activates a customer-specific BAdI, but first this is NOT relevant for the basic functions of planning via Allevo.
- On the other hand, it is important to specify the exchange rate type and the document type, which should be entered not only for plan asset line items for the plan year/version. As an example, here is an excerpt from the corresponding database table T895 with exchange rate type M and document type P0 (the recommended entries):

Planversion: Wechselkurstyp für ext. Datenübernahme pflegen

Ld	Ld-Bezeichn.	Ver	Bezeichn...	Jahr	BuKr	Name der Firma	Ktyp	B..	Beschreibung
0L	führendes Ledger	0		2012	1000	IDES AG	M	P0	Planung PC New GL
0L	führendes Ledger	0		2012	2000	IDES UK	M		

Figure 7: Database table T895

If the entry for the document type is missing, SAP displays the message "GLPL726 = Document type for plan line items missing or invalid".

If, depending on the SAP release, the above menu entry is missing in SAP Customizing, transaction SM30 with maintenance view V_FAGL_T895_KT can be called as a substitute.

- In order to work with plan line items, transaction FAGLGCLE must also be executed for the relevant years and versions (this call is also partially not defined in SAP Customizing SPRO).

Notice:	For migration to the New General Ledger, see also FAQs in SAP Note 1070629.

| | For posting statistical key figures in the New General Ledger (possible as of Allevo 4.0), see also SAP Note 1011595 - FAGLSKF: BAPI for posting statistical key figures |

5.8 Permissions

The authorization checks for the New General Ledger are carried out for each Profitcenter using the SAP function module *COTB_OBJECT_CHECK* (authorization object *K_PCA*), *co_action 1002*, *1003* and *3027 are* checked.

Authorization object *G_GLTP* is required for planning at company code level (balance sheet planning).

5.9 PC planning / reporting with separate Excel sheets per company code

Normally, PC planning always takes place at the level of a predefined company code (e.g. after selection by the user when entering the planning transaction).

As of Allevo 3.4, it is possible to transfer the company code flexibly from Excel to SAP. Application examples:

- a Profitcenter with several assigned company codes is to be processed via a single Allevo master.
- Reconciliation work between different company codes, which consists of a combination of PC and other object types.

The requirement can be realized via a MultiObject transaction, whereby the Allevo master contains several sheets with a fixed page structure (therefore also occasionally referred to as FISH variant for "FIx SHeets"). Each Excel sheet reflects the data of a company code. The corresponding abbreviation must be stored for each sheet in the cell with the area name CC_BUKRS:

- If this parameter is empty, Allevo will work with the company code as for entry, or from the PC master record if entry is made via a different object type.
- If the company code is invalid, an error message appears on SAP page.

This display variant can of course also be used very well for reporting functions.

| **Example** | Reporting can also be done on the level of PC groups, which are then to be entered in the master as a replacement for Profitcenters (for this, as always, the constant MULTI_WITH_GROUPS must be set). |
| | If Profitcenters of different company codes are stored in these groups, Allevo shows the company code-specific data for each sheet. |

5.10 PC planning in MO mode with delimitation of direct / indirect plan values

Some planning scenarios make it necessary to plan different object types simultaneously and from one planning form, e.g. a combination of Profitcenters and associated cost centers. This is possible with Allevo MultiObject.

The information about the objects involved is usually displayed in different rows. Since a Profitcenter usually serves as a parenthesis for other objects, it can be useful to display the directly posted values for the Profitcenter explicitly (separately from those from further allocation via other objects):

- When reading this data, the delimitation can be done via the relevant operations to the AllocationSet.
- When planning separate values, note that SAP planning transactions (e.g. 7KE1) always expect the total value. Allevo is also guided by this: if necessary, a summation of both shares must be carried out on the Excel side (direct and indirect).

Notice:	Allevo first posts the plan contents to CO objects. The corresponding values of the through posting are therefore already on the Profitcenter before the total value is to be posted there.
	This sequence ensures that in the end the total value is really on the Profitcenter (SAP internally checks the difference to the previous values and then creates the delta posting).

5.11 PC line item display and export

Allevo also supports the display of line items for Profitcenters (for the New General Ledger only from version 3.3.36).

The line item export via transaction /ALLEVO/SAT_LITEMS was initially only available for data for the Classic General Ledger (structure /KERN/IPP_S_PC_OFFL_ITEM_ACT with compilation of available fields). As of version 3.5, ,the /KERN/IPP_S_PC_OFFL_ITEM_NGL data structure is also available for the New General Ledger. The distinction is made in the Allevo layout via the PC_READTABLE constant (for details see Allevo SAP manual).

5.12 Allevo in the PC Reporting: Company code selection on entry

The original functions of Allevo were aligned with planning functions. With regard to the PC planning, this means that a company code must always be entered when accessing the planning (a prerequisite for posting plan data in SAP). See selection screen in Figure 1 with specification of the relevant company code. It must be ensured that only Profitcenters are read which are also assigned to the relevant company code. This standard behavior within planning can be set flexibly for reading reference data via the PC_REP_COMP_SEL constant (see F1 documentation).

If the Allevo is used for reporting purposes, the restriction to one company code is not necessary and also not always desired (especially if the relevant Profitcenters are assigned to several company codes). This additional flexibility is realized via the fixed value PC_REP_COMP_SEL with the following meaning:

1. If the constant is active, the company code can be selected directly when entering the Allevo Report transaction /ALLEVO/PCREP. The F4 possible value help there then shows the company codes for the current controlling area (in the standard case, only the set company code is displayed in this field).
2. Optionally, * can also be entered in the company code field in this case: Allevo then automatically totals the values across all company codes for display in Excel. This corresponds approximately to the behavior in SAP reporting (e.g. for PC plan/actual comparisons).

Notice:	In case (2) with sum over all company codes Allevo adds directly all values independent of the stored currency, which however can be different per company code. Therefore, currency type "Transaction currency" should be avoided, for example, as it can easily lead to incorrect results.
	See also constant NO_COMPANYCODE as an alternative to reading without a specified company code.

5.13 New General Ledger and S/4Hana in Allevo

Under S/4Hana, SAP recommends mapping Profitcenter accounting in the Universal Journal: in this case, the ACDOCP table is again relevant for planning. Activation takes place for each cost accounting area.

However, previous ERP tables for the New General Ledger are still available, e.g. FAGLFLEXT: now, however, as in the CO area as a so-called view. When querying these tables (e.g. in traditional SAP reporting), actual data is automatically derived from the corresponding Universal Journal table (i.e. ACDOCA). However, this does not apply to plan data, which can therefore only be evaluated using new SAP reporting functions (e.g. Fiori apps).

All of the planning functions can be helpful in this case, as the previously used interfaces to the new general ledger are still available (e.g. with posting to FAGLFLEXT). Consequently, this data is also available for target/actual comparisons in traditional SAP reporting.

Allevo also offers the option of transferring plan data to Universal Journal as a parallel posting (i.e. ACDOCP, as is also provided for in the CO area). For the associated settings for this S/4Hana interface, see the notes in the Allevo SAP manual.

6 Balance sheet report

6.1 Overview and activation

Allevo provides basic functions for a balance sheet report; usually as an output of values cumulated to the company code (similar to the presentation in SAP transaction "FC10" or "F.01").

Notice:	The transactions mentioned for SAP's own balance sheet report allow output via ALV. This data can therefore also be transferred to Excel via a satellite (see "Cockpit > Satellites > Transfer ALV data" or Allevo SAP manual). This means that the contents of the SAP report are available for further processing in Excel (if necessary, simply the balance sheet structure may also be helpful).

The desired evaluation structure can be permanently stored in the master; alternatively, a dynamic structure is also possible if, for example, the structure changes frequently or an Allevo master is to be usable for different balance sheet structures. A display example:

1000	Bilanzpositon	Bezeichnung	Ist -16 2011	in%	Ist -16 2012	in%	Plan -16 2013	in%
####	1032000	Sachanlagen	60.178.139	1,7	57.491.919	1,6	55.942.391	1,5
####	1033000	Finanzanlagen	11.455.784	0,3	12.392.881	0,3	12.392.881	0,3
####	1030000	Anlagevermoegen	71.633.923	2,0	69.884.800	1,9	68.335.272	1,9
300000	1041010	Rohstoffe 1	825.668.212	23,0	825.668.200	22,8	847.568.500	23,2
300010	1041010	Rohstoffe 2	61.625	0,0	61.625	0,0	61.625	0,0
303000	1041010	Hilfs- und Betriebsstoffe	762.859	0,0	762.798	0,0	762.798	0,0
304000	1041010	Ersatzteile	20.555.309	0,6	20.570.309	0,6	20.570.309	0,6
305000	1041010	Verpackungsmaterial	5.267.704	0,1	5.267.704	0,1	5.267.704	0,1
###	1041020	Roh-, Hilfs- und Betriebsstoffe	852.315.708	23,7	852.330.635	23,5	874.230.935	23,9
###	1041020	Unfertige Erzeugnisse, unfertige Leistungen	-408.194.437	-11,4	-380.385.861	-10,5	-377.817.826	-10,3
###	1041030	Fertige Erzeugnisse und Waren	2.571.740.166	71,5	2.572.918.266	71,0	2.573.623.666	70,4
###	1041040	Geleistete Anzahlungen fuer Vorraete			-937.097	-0,0	-937.097	-0,0
####	1041000	Vorraete	3.015.861.437	83,9	3.043.925.942	83,9	3.069.099.677	83,9
####	1042000	Forderungen und	23.085.705	0,6	26.901.621	0,7	6.618.351	0,2
####	1043000	Wertpapiere	-1.150	-0,0	-1.150	-0,0	-1.150	-0,0
####	1044000	Schecks, Kassenbestand, Bundesbank- und Post-	485.305.064	13,5	485.305.064	13,4	512.637.414	14,0
####	1040000	Umlaufvermoegen	3.524.251.056	98,0	3.556.131.478	98,1	3.588.354.292	98,1
#####	1000000	A K T I V A	3.595.884.978	100,0	3.626.016.278	100,0	3.656.689.564	100,0
####	2010000	Eigenkapital	425.314.134	-11,8	411.045.843	-11,3	392.889.979	-10,7
####	2020000	Sonderposten mit Ruecklage-Anteil	-2.294.611	0,1	-2.294.611	0,1	-2.294.611	0,1
####	2030000	Rueckstellungen	-11.947.693	0,3	-11.947.693	0,3	-11.947.693	0,3
####	2040000	Verbindlichkeiten	-4.006.943.395	111,4	-4.022.806.403	110,9	-4.035.323.825	110,4
####	2050000	Rechnungsabgrenzungsposten	-13.414	0,0	-13.414	0,0	-13.414	0,0
#####	2000000	P A S S I V A	-3.595.884.978	100,0	-3.626.016.278	100,0	-3.656.689.564	100,0
####	3010000	Umsatzerloese	-10.063.447	100,0	-3.457.621	100,0	-27.441.691	100,0
####	3020000	Bestands-Veraenderung	431.248.164	-4.285,3	-27.855.279	805,6	-30.979.599	112,9
####	3040000	Sonstige betriebliche Ertraege						
####	3050000	Materialaufwand	275.882.791	-2.741,4	500.623	-14,5	-4.636.066	16,9
####	3060000	Personalaufwand	25.134.034	-249,8	9.358.785	-270,7	18.825.605	-68,6
####	3070000	Abschreibungen	2.686.223	-26,7	2.686.220	-77,7	4.235.748	-15,4
####	3080000	Sonstige betriebliche Aufwendungen	5.609.058	-55,7	4.393.878	-127,1	8.449.560	-30,8
#####	3000000	Gewinn- und Verlust-Rechnung	730.496.823	-7.258,9	-14.373.394	415,7	-31.546.443	115,0
####	5011000	Ausserordentliche Ertraege	-0	0,0	-0	0,0	-0	0,0
####	5012000	Ausserordentliche Aufwendungen	6.277	-0,1	6.277	-0,2	6.277	-0,0
#####	5000000	Ausserordentliches Ergebnis	6.277	-0,1	6.277	-0,2	6.277	-0,0
####	6012000	Sonstige Steuern	52.893	-0,5	52.893	-1,5	83.728	-0,3
#####	6000000	Steuern	52.893	-0,5	52.893	-1,5	83.728	-0,3
####	7040000	Verlust-Vortrag aus dem Vorjahr	1.168.138.312	######	1.898.688.028	#######	1.898.688.028	-6.919,0
####	7050000	Gewinn-Vortrag aus dem Vorjahr						
####	7090000	Bilanzgewinn/Bilanzverlust	-416.239.361		-401.771.070		-383.314.137	
#####	7000000	Ergebnis-Verwendung	751.898.951	-7.471,6	1.496.916.958	#######	1.515.373.891	-5.522,2
#####	8000000	Nicht zugeordnete Konten	52.958.047		53.158.046		53.459.116	

Figure 8: Balance sheet report displayed in Excel

Some basic features: Balance sheet structure in the Excel frontend, determine balance sheet result specifically, expand hierarchy by double-click.

The procedure for activating the Allevo functions for the balance sheet report depends on the type of the general ledger used; a central control function is again performed by the PC_READTABLE constant (see also notes there):

- **New General Ledger** (as of Allevo 3.3.28)

  The reading of reference data to the New General Ledger is active if the table FAGLFLEXT is entered at "Value of" for the PC_READTABLE constant.

  In addition, "B" (= balance sheet report) must be entered in column "Value to" for this constant: as a result, the object stored in the Excel master is ignored and only the current company code is used for selection.

  By default, it is read with the ledger "0L"; deviating settings are to be stored in the column "Value to" at constant LEDGER.

  For reading actual data, version "1" must normally be entered in the TimeSet (instead of 0 as otherwise usual in CO).

- **Classic General Ledger** (from Allevo 3.4.23 and later)

  The relevant data (incl. actual data for balance sheet accounts) are in table GLT0 with key information about the company code: The table name must be entered accordingly at constant PC_READTABLE in column "Value from". Alternatively, it can also be the name of a customer-specific summary record table, if this is created with a data structure corresponding to Class. In this case, **G** must also be entered at "Value to" for the constant.

  The appropriate ledger is to be set via the constant LEDGER: via "Value to" as otherwise in the PC planning (usual are e.g. 00, L1, L2, without explicit specification also here 8A is the Allevo internal default).

  When reading actual data, pay attention to the appropriate version in the TimeSets (in many cases this is version 1).

  Usually, reading should be done via the company code currency (so. Allevo currency type O = object/local currency). If alternatively reading is done via group currency (currency type C = controlling area currency), the update of the corresponding KSL fields in GLT0 must be ensured via SAP customizing.

Depending on the version, Allevo also checks the status of the representative objects during reading; in this case it may be useful to use the constant STATUS_READ_ALL.

In special cases, it can make sense to display debit and credit postings via separate AllocationSets in Allevo: for this, constant PC_BEKNZ must also be set. However, if additional planning functions are used, this can easily lead to aggregating postings.

In the standard case, Allevo checks whether company code and controlling area are assigned in SAP Customizing (this can be switched off for the balance sheet report via MESSAGE_LEVEL).

Notice:	If the entry of plan data for the balance sheet report is not done via Allevo, the following SAP transactions are available: for the Classic General Ledger, the entry of plan data is normally done via the transaction FSE5N (standard profile SAPFIGL is stored). Transaction GP12N, which is also intended for the New General Ledger, is more flexible.

6.2 Company codes as characteristics of the selection

In both cases (classic and New General Ledger) the company code is the relevant selection criterion. For the determination:

- When reading reference data, Allevo adopts the relevant company code as stored on the Excel side for the current sheet in the standard case; this corresponds to the entry in the "Local Information" on the "Info" sheet (or CC_BUKRS in older master versions). These entries are usually preset with the company code of the entry object, but can be changed on Excel side before calling the read function. Advantage and example: when called in the MultiPage mode it can be useful to read postings for different company codes and if necessary to plan them again (separated by sheets, see also "Call Allevo" below).

- A special representation is possible when entering via the Allevo object as representative of a company code. In this case, the company code can be taken from the object of the row (which corresponds to the KX object) using the NO_COMPANYCODE constant with the option K. In this case, different company codes do not have to be displayed on different sheets (e.g. display in SummaryCorner is possible).

Notice:	Brief technical background: the company code of a sheet is transferred from Excel to SAP via the Allevo header data (OBJECT6 field). If in doubt, simply use TAB2SAP to display the transfer tables.

6.3 Additional features of the selection

In both cases (classic and New General Ledger), the company code is the central selection characteristic. Optionally, Allevo can also take other characteristics (such as functional area) into account in the data selection (see notes on reading about additional characteristics in section 3.3 above).

Reading via additional characteristics can also be used to put postings of different company codes on one sheet.

Up to Allevo 3.5, the following procedure applied: the company code itself is provided with the help of the READ_WITH_KEYR constant (* must be entered for Local Infos or CC_BUKRS). In this case, the Excel side uses the entry in the CY_KEYR column to decide which company code is to be read. Depending on the case, it is also recommended here to access via the Allevo object with the same abbreviation as for the company code (entry can then be transferred to column CY_KEYR, for example).

As of Allevo 4.0, pointers are to be used according to the relevant field name in GLT0.

6.4 Call via Allevo Layout

You can access Allevo via a representative Profitcenter: data is read for the company code that is stored in the master record for the Profitcenter.

In practical applications, it can also make sense to access via an Allevo object per company code, whereby the company code must again be entered in the master record for transfer to Excel. The transition to object type PC takes place in the Allevo master, therefore the call must be made in MultiObject mode. Advantages of calling up via Allevo Object:

- Via the category for the Allevo object, an individual field designation is offered in the Allevo initial screen (e.g. "Company code").
- Via the F4 value help for the Allevo object, the user automatically receives a selection of the relevant company codes.

In order to display data from different company codes in the same Allevo Master, a MultiPage variant is useful in many cases, whereby the company code must be unique in the header of each page (see SheetObject area, formerly CC_BUKRS).

6.5 Transfer balance sheet structure dynamically to Allevo Master

6.5.1 Overview

The structure of a balance sheet report can be permanently stored in the Allevo master. Alternatively, a dynamic master structure is also possible, which is based in each case on the balance sheet/P&L structures currently stored in the SAP system: This can be advantageous, for example, if these structures change frequently or if an Allevo master is to be usable for different balance sheet structures.

For the second form of display, Allevo offers support via several functions.

6.5.2 Read balance sheet structure for transfer to Excel

The transaction /ALLEVO/EXP_BAL_STR allows to read any SAP balance sheet structure in order to transfer it to Excel via satellite. Selection parameters are controlling area and abbreviation of the balance sheet/P&L structure. This procedure is suitable if changes to the structure are rare, but the structure is not to be stored permanently in the Allevo Master.

The transaction displays any balance sheet structure (e.g. for GKR = commercial balance sheet Germany) with all relevant details as an ALV list. The hierarchy levels are displayed using special characters, as usual for Allevo, especially also in MOD functions (for structure, see ALV output structure /KERN/IPP_S_BALANCE_STRUCTURE). The most important components:

Component	Kuzr designation	Notes
OBJECT	Planning object	currently not relevant

ERGSL	Balance sheet/P&L item	Position number, for rows with accounts it is the parent position
SAKNR	G/L account number	G/L account or Allevo hierarchy level (e.g. '###')
ASSET_CATEGORY	Category	Assets or liabilities (0: none; 1: assets; 2: liabilities)
ASSET_TYPE	Type	Keys correspond to the entries in table T011 and thus to the "special items" in transaction FSE2 for maintaining the balance sheet and P&L structure: 1 = Assets 2 = Liabilities 3 = Accumulated deficit 4 = Retained earnings 5 = P&L profit 6 = Unallocated accounts
GVTYP	Profit and loss account type	Like entry in SKA1-GVTYPE
JUMPING	Jump license plate	0 = Normal, 1 = Jump group/saldo change position 2 = Jump account/saldo change account
JUMPING_ERGSL	Balance position/ Jump group	Corresponding balance change position/jump group, if available
HS_INDICATOR	Debit/Credit Indicator	Debit (S), Credit (H) or both (A)
STAGE	Stage	Level of item in the balance sheet/P&L structure (assets/liabilities = 1)
REG_TYPE	Indicator	Account (A), position/total row (S)
OBJTEXT	Designation object	currently not relevant
ACCTEXT	Designation account	Name of the account

To transfer the data to a satellite, please use the usual ALV transfer function of Allevo. To simplify the application, there is the GroundTable /KERN/U_BAL06_A with all the fields provided, which basically replaces an MOD satellite completely.

6.5.3 Provide balance sheet structure via DynamicSet

As an alternative to the transaction /ALLEVO/EXP_BAL_STR, as of version 4.1 Allevo offers the option of rebuilding the balance structure via a DynamicSet each time the Allevo report is called up. The corresponding type is BS (= Balance Structure); the abbreviation of the relevant balance structure is given directly in the DynamicSet as dummy filter DM. Example:
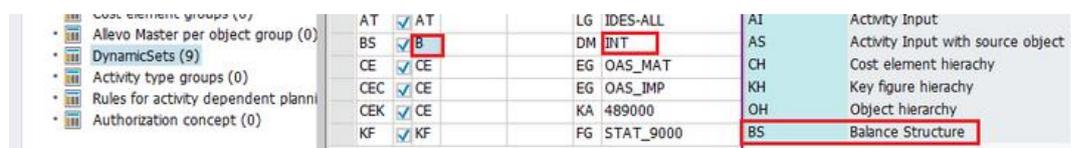
Figure 9: Provide balance sheet structure via DynamicSet

The most important fields to pass to Excel (equivalent to the components in the previous section):

Fields	Short name	Notes
ALLOCATIONSET	Allocationset Read	Found AllocationSet for use with the DialogCorner
ALLOCATIONSET_POST	Allocation posting	Found AllocationSet for use with the DialogCorner
OBJECT	Planning object	currently not relevant
OBJECTTYPE	Object type	
OBJTEXT	Designation object	
FIN_STA	Balance sheet/P&L item	For accounts assigned to the parent item
ELEMENT	G/L account number	G/L account or Allevo hierarchy level (e.g. '###')
ASSET_CAT	Category	Assets or liabilities (0: none; 1: assets; 2: liabilities)
ASSET_TYPE	Type	Keys correspond to the entries in table T011 and thus to the "special items" in transaction FSE2 for maintaining the balance sheet and P&L structure: 1 = Assets 2 = Liabilities 3 = Accumulated deficit 4 = Accumulated profit 5 = P&L profit 6 = Unallocated accounts.
PL_ACC	Profit and loss account type	See SKA1-GVTYPE
CONTRA_ITEM_X	Jump license plate	0 = Normal, 1 = Jump group/saldo change position 2 = Jump account/saldo change account
CONTRA_ITEM	Balance position/ Jump group	Corresponding balance change position/jump group, if available
HS_INDICATOR	Debit/Credit Indicator	Debit (S), Credit (H) or both (A)
LEVEL	Stage	Level of item in the balance sheet/income statement structure (assets/liabilities = 1)
REG_TYPE	Indicator	Account (A), position/total row (S)
ELEMENT TEXT	Designation	Name of the item or account. As the structure is read dynamically, the texts of the balance sheet structure are displayed in the logon language.

6.6 Authorization check

Equivalent to the behavior in the SAP transaction GP12N, Allevo checks the authorization object G_GLTP = "Special ledger - database (ledger,record type,version)" and additionally F_FAGL_LDR = "general ledger: authorization for ledger" (each with check on plan and actual data) when reading data.

6.7 Line items in the balance sheet report

The jump to the line item view, for example, is not yet available. In individual cases, it may be useful to transfer data from an equivalent SAP report to a satellite (e.g. from transaction FAGLL03).

6.8 Balance carried forward / inventory values

Balance sheet accounts show key date values that are composed of a balance carried forward during the year (e.g. GLPCT -HSLVT Balance carried forward local currency / period from year-end closing) and the current flows in the months.

Notice:	Allevo recognizes stock accounts by the "Indicator: Account is stock account? in the G/L account master record (field SKA1-XBILK).

Allevo behaves as follows when reading reference data:

- Monthly columns contain only the respective, month-related movements.
- TimeSets for years also contain the carry-forward (is therefore read automatically). As of 4.0, only read if the period 0 is also entered in the TimeSet (value of).

There are two ways to explicitly read the balance carryforward:

1. A TimeSet is created explicitly for the previous year and included accordingly (Balance sheet value = Carry forward + Monthly valuation). See the following example.
2. As of version 4.0, Allevo allows a special period "0" in the TimeSet to explicitly address the talk (a specification "from Period 0 to Period 0" accordingly only reads the carryforward).

Example with reading of stock values according to variant (1) in the transaction currency:

- Table field TSLVT contains the start balance of the year.
- Period-related transaction data are in the fields TSL01, TSL02, ... TSL16, which are read into the Allevo TimeSets.
- The carry forward (as annual value) is read via an additional TimeSet with reference to the previous year.
- This value then corresponds to the entry in the associated VT field "TSLVT".

6.9 Control for assets/liabilities

A balance sheet can contain accounts that have either a negative or a positive balance (to be assigned twice in the chart of accounts accordingly). The sign decides whether the value/account must be shown in the assets or liabilities area.

As of Allevo 4.0, the asset/liability distinction can be controlled by the "Value Filter" feature in the master record for the allocation type. The following are available for selection:

1. Hide for negative values
2. Hide for positive values

On the Excel side, the relevant accounts are usually stored twice: both in the assets and in the liabilities. Depending on the current value, one of the two rows is not supplied with values and then automatically hidden in Excel.

Notice:	In earlier Allevo versions, control was via the hidden constant "ROWKEY_BALANCE_PC (see manual 3.5).

7 Balance sheet planning / planned balance sheet

7.1 Overview

Balance sheet planning with Allevo is currently always carried out at company code level (segment level is currently not supported). Allevo enables the entry of plan data via the familiar Excel interface and supports its transfer to the New General Ledger (table FAGLFLEXT) and the Classic General Ledger (usually table GLT0).

Whether the entry is made via a master that is intended for balance sheet reports should be clarified in the implementation project. Often, posting the Allevo module "FP Flexible Planning" is also useful (see also notes below on the setup for the new and the Classic General Ledger).

In addition to the direct input of planning data, Allevo also offers a range of functions for importing data from upstream (specialist) planning. Individual examples of this:

- Very often, detailed information with suitable allocation of accounts and organizational units is already available via Allevo specialist planning (e.g. via investment or personnel planning to cost center).
- Cost planning could, for example, have been carried out in detail at cost center level.
- Sales could have been planned in COPA, at Profitcenter level, via internal orders, or even in external systems.

Of course, many other constellations are conceivable. What they all have in common is the idea of viewing existing plan data as the basis for subsequent balance sheet planning. In accordance with this idea, Allevo offers a set of functions for reading data from the corresponding tables and preparing it in a suitable form. The following graphic shows the basic structure of the solution:
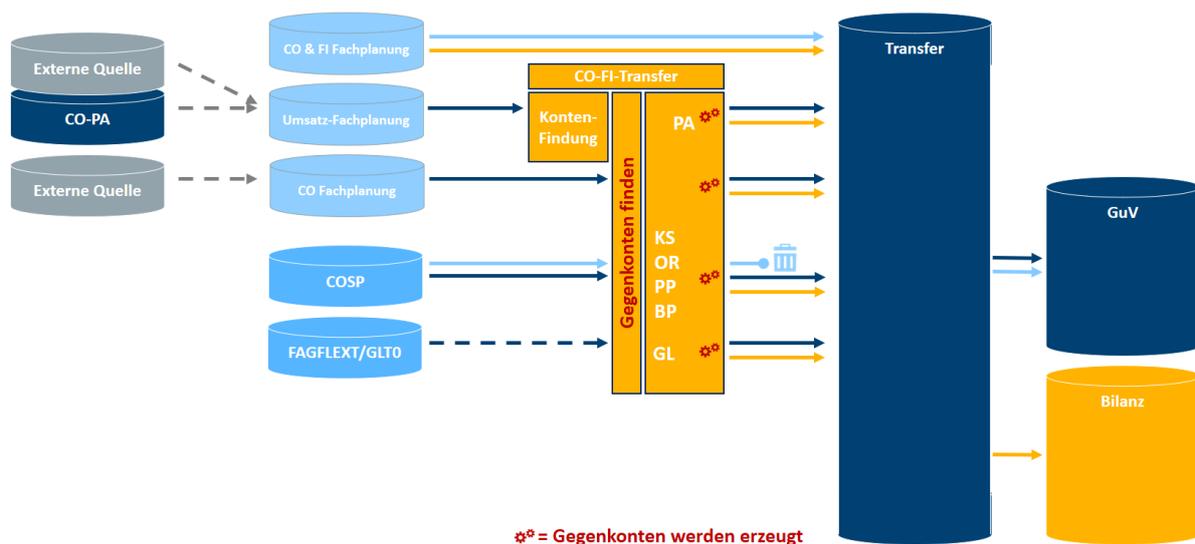
Figure 10: Balance sheet planning with data transfer from preliminary plannings (CCA,OPA,PA,satellites...)

Allevo provides its own transactions for the individual data transfer functions, which are described in the following sections. The central component of the system is a transfer table in which all data is

collected, prepared and further processed according to priorities (a satellite with specific characteristics, delivered via as a GroundTable).

The end result are performance characteristics like these:

- Balance sheet and P&L planning based on reliable and complete planning data
- Automatically assign balance sheet postings to profit and loss postings and include them in planning according to the rule "Posting = offsetting entry, balance 0".
- Analytical cash flow planning based on bank accounts, receivables, payables, etc.
- Excel based balance sheet reporting (see previous chapter)

In particular, the "Balance 0" target is taken care of by the above-mentioned formatting functions for the transfer table with meaningful rules. Keywords:

- Receivables correspond with revenues
- Material costs are followed by liabilities
- For personnel costs there are the offsetting entries at the bank
- Intercompany business thrives on partner postings
- Depreciation reduces the asset portfolio
- Payments received reduce receivables
- Liabilities decrease with outgoing payments ...

7.2 Transactions for transferring data from upstream planning

7.2.1 Steps for transferring data from preliminary planning

According to the list above, Allevo offers transactions to read data from the following tables and make it available for balance sheet planning:

- Allevo satellites with any specialized planning
- Totals record tables for SAP CO (COSP) and Allevo totals record tables (ACOSx)
- general ledger summary record tables (e.g. FAGLFLEXT)

Figure 11: Program for transferring costs/revenues to the balance sheet planning

The associated transfer programs read data from these source tables in order to collect them after intermediate steps in a so-called transfer table: this is again a satellite, which allows further additions or corrections if necessary. In the transfer table the source is noted as well as a priority: in this way e.g. multiple entries to an account from different sources are not necessarily disturbing, because the further processing depends on that source, which was provided during the data transfer with the highest priority.

Data is posted using Allevo's own functions in the Flexible Planning module (see also notes below on setting up the new and the Classic General Ledger).

In the step before the posting, Allevo offers a special transaction to prepare the source data of the transfer table appropriately (e.g. for finding and supplementing contra account or also VAT rates). Currently this function is bound to /KERN/U_BAL05_A as transfer table and further GroundTables for the preparation of the data (/KERN/U_BAL02_A to /KERN/U_BAL04_A).

7.2.2 Transfer data from Allevo specialist themes (satellites)

Very often, detailed planning data is already available via Allevo specialist planning, which can be used as the basis for balance sheet planning (including suitable assignment of accounts and organizational units). Examples could be investment or personnel planning on the basis of cost centers.

The corresponding transaction is /ALLEVO/GL_READ_SAT. The transfer of data of a satellite table is done in these steps:

- Read the source table according to the desired selection
- Preparation of data via built-in functions (e.g. suppression of blank values, account determination, sign reversal...)
- Display of the prepared data in an ALV list (if necessary also for checking)
- Transfer to the transfer satellite via a defined field mapping (the standard Allevo ALV transfer is used).

Consequently, the data transfer setup consists of two technical steps:

1. Definition of the selection parameters and the desired formatting functions in the transaction is /ALLEVO/GL_READ_SAT (stored as a report variant).
2. Definition of the mapping via a corresponding schema in transaction /ALLEVO/RD_PRG_ALV with call of the report variant from (1).

The two steps are explained in detail below.

Definition of the selection parameters and the desired formatting functions

The transaction /ALLEVO/GL_READ_SAT provides the following parameters in the initial screen:

Figure 12: Balance sheet planning with transfer of data from the specialist planning

The first section defines the selection conditions to the source satellite; primarily defined by the default fields of a satellite:

- The transfer of data always takes place within one controlling area and data of one year or version.
- Normally, a separation by object type is also useful (e.g. costs for the cost center, but sales for orders).
- In individual cases, a very specific selection query can be made using the "Additional selection" field; here, specific fields in the satellite can be explicitly taken into account as in an ABAP Where condition (please coordinate the syntax with our support if required). Use case: Data in the satellite should be transferred in different program runs depending on certain characteristics.

The other options support data preparation before transfer to balance sheet planning. Currently, the following functions are supported (representation in the initial screen also corresponds to the processing sequence):

- **Query for empty records / rows without values**

  This option can remove rows that do not contain meaningful data. For this, the relevant value fields are queried on initial; the default of these fields in the satellite is done via a multiple selection.

- **Determine partner grouping / Find EU country**

  The classification of a posting row according to the criteria INL (= Domestic), EU (=European Union) and ROW (= Rest Of World) can be helpful for a subsequent assignment of a G/L account. Allevo uses the SAP master data for countries in the table T005 for this purpose. As a basis for this

function, three fields must be available in the satellite: the leading country and that of the partner as the basis for the determination, as well as a field for the determined abbreviation of the classification ("Target field EU country determination").

- **Account determination parameters**

  If no accounts have been assigned within the specialist planning, an automatic determination via additional characteristics in the satellite can be helpful. Application example: the data in the satellite originate from postings for imputed COPA.

  The basis for the determination is another satellite with assignment of various characteristics to an account. The ground table /KERN/U_BAL01_A, for example, which is supplied in the Allevo standard (with a selection of characteristic fields, which is particularly useful for account determination with common COPA data), can serve as the basis. Account determination therefore requires a reference to this "determination satellite" in the first step.

  In the following, an assignment of characteristics is required that should lead to the matching account: in each case with specification of the field in the finding satellite and in the current satellite whose data is currently being processed. Allevo can currently consider up to 7 features in the finding process: first an account is searched for at the highest level of detail, then with the initial value of the last feature, and so on until the search is successful.

  Of course, the detection also requires a mapping to the target field of the detection; in this case, the account that is to be transferred to the satellite structure is the first priority. In addition to the account, however, other properties can also be "found" (e.g. for rule-based sign reversal via a factor): Allevo currently supports three of these target fields.

- **Value date year >> period**

  It can be useful to assign a value to a period when transferring a value that was originally entered for a year to balance sheet planning. To do this, Allevo requires an entry for this period (or the field name in the satellite), the field name with the value specification, and a prefix for the period field in the satellite (e.g. the abbreviation "VALUE" if the annual value is to be transferred to a field between "VALUE01 and VALUE12").

- **Change values by the factor**

  Allevo can factor any values from the source satellite. The most common application is certainly a sign reversal (i.e. multiplication with the factor "-1"). The factor must be specified via a field in the satellite: if necessary, it can also come from the account determination as already described above. The relevant fields of the conversion have to be entered again via the multiple selection.

As a result, the report delivers an ALV list with all relevant posting rows, which are transferred to a transfer satellite via Allevo standard functions of the ALV transfer, if necessary. A suitable mapping is then also to be set up there via standard functions (including assignment of a field with information on the source and priority). The report here therefore does NOT create the transfer satellite itself, but is only used for data selection and preparation.

Notice:	It makes sense to store each data transfer with its own Allevo object in the transfer table. This ensures that no contents from other program runs are overwritten. If necessary, the correct, previously determined data will also be replaced when the program is run more than once.

7.2.3 Transfer data from (SAP) summary record tables

Background: A cost planning is usually carried out in detail at cost center level. It would therefore make sense to also use the associated plan data as a basis for the balance sheet planning.

Allevo supports e.g. the CO object types KS and OR as well as a data transfer from Profitcenter planning. The corresponding transaction is /ALLEVO/GL_READ_GEN. The necessary steps for a data transfer:

- Determination of source and currency (e.g. for reading primary costs from COSP).
- Read the source table according to the desired selection
- Display of the prepared data in an ALV list (if necessary also for checking)
- Transfer to the transfer satellite via a defined field mapping (the standard Allevo ALV transfer is used).

Consequently, the data transfer setup consists of two technical steps:

1. Definition of the selection parameters and the desired formatting functions in transaction is /ALLEVO/GL_READ_GEN (stored as a report variant).
2. Definition of the mapping via the corresponding schema in the transaction /ALLEVO/RD_PRG_ALV with call of the report variant from (1).

The two steps are explained in detail below.

Definition of the selection parameters and the desired formatting functions

The transaction /ALLEVO/GL_READ_GEN provides the following parameters in the initial screen:

Figure 13: Balance sheet planning with transfer of data from (SAP) summary record tables

The current intended data source is:

- The SAP table COSP with primary costs can be read for the object types KS and OR; optionally, values can be transferred in the object or controlling area currency.
- If the original entry of the plan data was made using Allevo's own functions, it may be useful to transfer from table /KERN/ACOSP with the same selection options.
- For Profitcenter planning data in the New General Ledger, a transfer from table FAGLFLEXT is provided (either in group or local currency).

Depending on the tables mentioned, the characteristics of the selection in the initial screen may also change.

Zero rows (i.e. no values in the respective currency) are automatically suppressed.

In addition, Allevo aggregates the data before displaying it as an ALV list, ignoring characteristics that are not relevant for subsequent balance sheet planning functions anyway (optionally also without debit/credit indicators). The log at the end is intended in particular for test purposes and shows, among other things, the number of rows before and after the aggregation.

As a result, the report delivers an ALV list with all relevant posting rows, which can be transferred to a transfer satellite via Allevo standard functions of the ALV transfer, if necessary. A suitable mapping is then also to be set up there via standard functions (including the assignment of a field with information on the source and the priority). The report here therefore does NOT create the transfer satellite itself, but is only used for data selection and preparation.

Notice:	It makes sense to store each data transfer with its own Allevo object in the transfer table. This ensures that no contents from other program runs are overwritten. If necessary, the correct, previously determined data will also be replaced when the program is run more than once.

7.2.4 Prepare data of the transfer table

Background: via the upstream transactions described above, data is read from a wide variety of sources and collected in the transfer table. The transaction /ALLEVO/GL_PREP_POST provides functions to prepare the source data of the transfer table appropriately before transfer to the balance sheet planning: this includes the assignment of tax accounts and receivables/payables accounts (partner/counterpart accounts) via distribution rules. In the last step, the posting is made via the FP Flexible Planning.

Tax accounts are determined via the Partner Manager, as well as receivable/payable accounts (partner/counterpart accounts) with distribution rules as a basis for the balance sheet planning.

Currently, this function is bound to /KERN/U_BAL05_A as a transfer table and other GroundTables for formatting the data (/KERN/U_BAL02_A to /KERN/U_BAL04_A). The parameters in the initial screen:

Figure 14: Balance sheet planning with preparation of data from the transfer table

The result of the formatting is written to a selectable target version. For test purposes, a display function is provided: two ALV lists appear one after the other, displaying the originally selected source data and the result of the formatting (determination of contra accounts).

Determination of tax, receivables and payables accounts

The basis of the determination are so-called "partner groups", which are determined via the table /KERN/U_BAL02_A (based on company code, object type and account: if necessary with any "detail key" to map individual differences depending on organizational units).

On the basis of a partner group, Allevo determines so-called "partner rules" for the percentage distribution of the current posting (therefore also called "splitting"): the rules are stored in the table /KERN/U_BAL03_A. In a simple case this would be e.g. two additional rows with the shares for the contra account and value added tax. Both are mapped via another characteristic "Partner-Key". Here also a distribution of annual values to months can be stored.

Depending on the stored tax key, Allevo reads the appropriate tax rate directly from SAP tables (tables A003 and KONP-KBETR). Depending on the amount (= sum over all periods) Allevo also sets the debit/credit indicator.

In the last step (the so-called "partner determination") the partner key allows the determination of suitable accounts via the table /KERN/U_BAL04_A: credit/debit accounts and corresponding tax accounts are stored there.

If the characteristics are not completely maintained in the determination process, Allevo generates an error message and transfers this to the ALV output: it is therefore directly traceable where master data may still need to be maintained.

7.2.5 Cash manager for reduction of receivables and payables

Background: if a payment history is known and can be maintained in master data, a significant amount can already be done automatically when reducing receivables and payables, leading to further time savings in the balance sheet planning.

Allevo supports this automatic reduction of receivables and payables. However, the reduction of opening balances must continue to be planned via the CashManager, whereby the planner can focus on corrections and plausibility checks.

Figure 15: Balance sheet planning with preparation of data from the transfer table

The CashManager function runs through the transfer table to process the previously created receivable/payable accounts. Splitting and timeshift rules are found based on a master data table. Functions:

- Allocation account percentage with shift by x months into the future
- Creation of an offsetting position for specified bank accounts (specified in the master data)
- Hedging, so that the months postponed beyond December appear in the following year (if necessary, also subsequent version).

7.3 Balance sheet planning in the New General Ledger

7.3.1 Function overview

Planning is done using the same functions (same BAPI) as already described within Profitcenter planning. Consequently, we will mainly deal with special features here.

The control of the planning functions via table /KERN/IPP_PCNGL allows posting in the New General Ledger even without specifying the Profitcenter (see, for example, row 4 of the Figure 5). This results in the possibility of transferring planning data at company code level, equivalent to transaction GP12N.

| Notice:	Calling GP12N can also be useful for comparisons. The relevant parameters when calling the SAP transaction (in the default case):
	- Planner profile SAPFAGL
	- FAGLFLEXT summary table
	- Layout 0FAGL-07 (account / balance sheet planning).

In Allevo, planning transactions are always accessed via a specific object type, e.g. by entering a cost center or Profitcenter. However, the company code is not currently implemented as an individual "object type"; consequently, the company code should be derived from another (representative) object. The following variants are available:

- Access via the Profitcenter with matching company code in the master record.
- Entry with the Allevo object representative of the company code (for setup see notes below)
- The company code is determined individually in Excel and made available via the header data of the Allevo interface (see SheetObject area or CC_BUKRS in previous Allevo versions).

In Allevo balance sheet planning, the same constants are relevant as in the PC planning (e.g. PC_NGLA_PLANNING); constant PC_READTABLE must be set to "FAGLFLEXT and contain a "B" in the column "Value to". For details see the F1 documentation on constants.

The authorization check on company code level is done via the authorization object G_GLTP. When accessing via a representative Profitcenter, the check for the respective Profitcenter is skipped.

Here is an example with matching entries in the table /KERN/IPP_PCNGL with input via the transaction /KERN/IPP_PC_NGL.
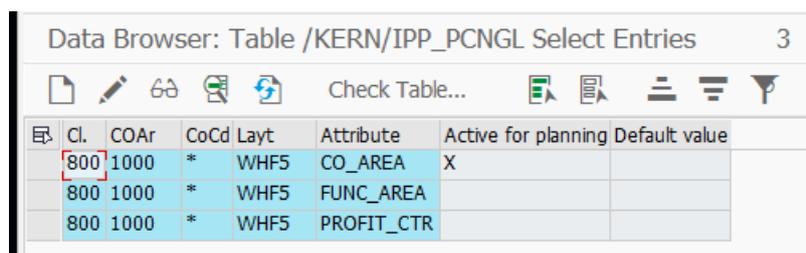
Figure 16: Definition of characteristics when posting to the New General Ledger

In particular, it is necessary to enter the optional parameter PROFIT_CTR and to deliberately switch off the active flag. This ensures that no Profitcenter is transferred. To have a clear definition, it is recommended to work with explicit layout names in this table and not to make * entries for the layout. The layout is valid independent of the object type (i.e. also for entry via KX, but for the entry in the transaction /ALLEVO/PC_NGL a dummy layout for the object type PC must exist).

Notice:	If an invalid characteristic is transferred to the SAP BAPI, message KM2005 appears.

On the Excel side, PC must therefore be entered as the object type (MO_AOTYP), although in the strict sense no Profitcenter planning takes place: controlled via the above-mentioned settings, Allevo automatically switches internally to the company code level.

Stock accounts: no cumulative plan data entry

The SAP BAPI can only post monthly values. If annual values are transferred from the Excel side, Allevo automatically divides the values linear among the months. This also applies to inventory accounts.

For this reason the function "Cumulative plan data entry for balance sheet accounts" in SAP NGL Customizing should not be active when working with Allevo, it currently leads to different results than in GP12N.

Parallel plan data entry to Profitcenter is not supported

If plan data for a combination of year and version is entered at an organizational characteristic level (e.g. company code), then no plan data may be entered in parallel (e.g. using a different layout) at another characteristic level (e.g. with specification for the profit center).

Background: Allevo would also read the PC level data at company code level and rewrite the aggregated value during planning (and thus ultimately increase the value in each planning operation). Correct would be a difference posting, but this is not supported by Allevo. The only way is a different version for the two posting transactions.

7.3.2   Balance sheet planning when entering with the Allevo object (KX)

In practical applications, it is advisable to use one Allevo object per company code; the company code must again be entered in the master record for transfer to Excel. In this case, the transition to the PC object type must take place in the Allevo master, so the call can only be made in MultiObject mode.

Advantages when called via the Allevo object:

- Via the category for the Allevo object, an individual designation can be offered in the Allevo initial screen (e.g. "Company code").
- Via the F4 value help for the Allevo object, the user automatically receives a selection of the company codes for which planning is relevant.

Special features for access via KX:

- For collecting entries in the Allevo table /KERN/IPP_PCNGL a Profitcenter layout with the same name must currently exist (normally only the layout abbreviation is required without other settings).
- Planning via account groups (see note above) is possible if the corresponding 1:N entries for the PC layout with the same name are available. The PC layout only has to be created with an abbreviation and a description: Properties such as constants or TimeSets always result from the initial layout.
- In the master, PC must be defined as the object type in column MO_AOTYP (although no Profitcenter postings are actually made.
- In current Allevo versions, the entry object can be used representative of the company code also in Excel column MO_AO. A check for a Profitcenter master record (as is usually the case with object type PC) no longer takes place. In previous Allevo versions (before 3.5.45), a dummy Profitcenter had to exist and the corresponding status had to be set in the PC layout.
- The status is only updated to the KX entry object. For reading reference data, the constant STATUS_READ_ALL must be set to "3", otherwise the message appears that a valid status for the PC layout was not found.

7.3.3   Application in the Allevo module "Flexible Planning (FP)"

As of version 3.5.8, balance sheet planning can also be carried out via FP. A suitable Allevo layout must be assigned in order to evaluate the relevant constants. "PC" must also be entered here as the OBJECT_TYPE. The FP constant FD_RECEIVER has no meaning in this case; it makes sense to also assign the company code here (so the same mapping as for FD_RCV_CCODE).

7.4   Balance sheet planning in the Classic General Ledger

7.4.1   Overview of realization in Allevo

Since version 3.4.28, Allevo offers functions to also enter data for balance sheet planning for the Classic General Ledger: functions of SAP transaction GP12N are executed in the background for this purpose (Batch Input).

Allevo supports the entry of annual and monthly values with the following restrictions:

* Data is recorded at company code level without additional characteristics such as business area.
* It is not possible to plan quantities.

The associated SAP table for the Classic General Ledger is GLT0. The Allevo function for balance sheet planning is automatically active if GLT0 is set up for reading (constant PC_READTABLE). Further parameters:

* The relevant object type in the Allevo master is still "PC".
* Usually, a planning object is also stored in the Allevo master for each posting row; however, it is ignored by the Allevo planning functions if the constant PC_READTABLE is active for GLT0 (so any dummy Profitcenter can be entered in Excel).
* If planning takes place at the level of account groups, these must be stored in the usual Allevo 1:N master data. Important: when entering via an object type other than PC (e.g. via an KX object), the account groups must be entered via an Allevo PC layout with the same name).
* The ledger must be entered at constant LEDGER ("value to", e.g. 00, L1).
* Constant PRE_SELECT must not (!) be active.
* All plan values are transferred to SAP; Allevo therefore ignores settings for NO_ZERO_DELTA (to have the behavior transparent, constant should also not be set).

Posting procedure

For posting the plan values, Allevo uses a batch input to transaction GP12N:

* Allevo automatically selects the SAP standard planning profile SAPFIGL (= planner profile for FI-GL planning) and for the table GLT0.
* The input is done via layout 0GLT0N1, the first view in the transaction GP12N.
* The same authorization checks are run as when calling GP12N in the dialog (e.g. check for authorization object G_GLTP).

Notice:	Before posting via Allevo, you should ensure that the relevant planning periods are activated for each company code (see transaction GCP5). If necessary, it may be useful to first create test postings manually via the transaction GP12N to check the prerequisites.
	Important: To enable the batch input on GP12N, the Excel integration must NOT be active for this transaction. This setting is stored in SAP Customizing for the planner profile (here SAPFIG and layout 0GLT0N1).

In the standard case, Allevo takes the relevant company code from the header data of the Excel side (e.g. CC_BUKRS): in this case, all data of an Excel sheet must refer to the same company code (which is then usually determined via the object of the sheet). Consequently, Allevo must be called up in MultiPage mode if data for several company codes is to be displayed in the same Allevo master.

However, via additional setup steps, the company code can be copied from the rows of the planning area on Excel side (see section 4.5).

To address customer specific special ledger tables (instead of GLT0) see section 7.4.4.

7.4.2   Allevo planning call

Allevo can be accessed via a representative Profitcenter: in the standard case, data is then read for the company code that is stored in the master record for the Profitcenter. In the first step, the company code is transferred to the LocalInfos in the Allevo master (or previously CC_BUKRS), the parameter is retrieved via the header data and is then the basis for the subsequent functions for reading and planning. In this way - depending on the use case - functions on the Excel side can also change the relevant company code.

In the practical application case, it is recommended to access via an Allevo object per company code, which must be entered there again in the master record for transfer to Excel. In this case, the transition to the PC object type must take place in the Allevo master, which means that the call can only be made in MultiObject mode.

Advantages when called via Allevo object:

- Via the category for the Allevo object, an individual designation can be offered in the Allevo initial screen (e.g. "Company code").
- Via the F4 value help for the Allevo object, the user automatically receives a selection of the company codes for which planning is to be carried out.

When entering via an Allevo object (KX), please note the following special features:

- On the Excel side, the object type PC must still be entered and consequently an existing dummy ProfitCenter must be entered as an object (but in the end without any controlling effect).
- For the respective dummy ProfitCenter, the status must be set in a PC layout with the same name; alternatively, constant STATUS_READ_ALL = 3 can be used.
- For planning via an account group (see note above) 1:N entries to the PC layout with the same name may be required. For this, the layout does not have to have any further properties (only master record with the abbreviation). Properties from constants or timesets always result from the initial layout.

- In current Allevo versions, application of dynamics (e.g. MOD) is also possible.

7.4.3 Currency conversion

Allevo currently only uses the form-oriented entry of the transaction GP12N: there, the transfer of a currency key is not possible, or is prevented after the first posting. Instead, the posting is always made with the currency that is stored in the master record for the company code (as is also the case when GL12N is called up onrow).

Consequently, the currency type of an Allevo TimeSet has no effect on planning; however, it should be set correctly for TimeSets for reading reference data (normally O = company code currency).

The GP12N internal translation to controlling area currency is performed according to the rules defined in SAP Customizing.

Special case: currency unit in master record for account

If a specific currency unit is defined in the master record for the account, it is also binding for the posting in GP12N (cannot be changed there).

However, via Allevo, plan data is usually entered in a uniform currency (e.g. the company code currency). In this case, Allevo takes into account the entry in the master record for the account and converts to this currency before transferring the values to GP12N: this logic currently only applies to balance sheet planning for the Classic General Ledger. The following applies:

- The relevant exchange rate type is taken from the PCA table T895PCA (= fiscal year-dependent version parameters).
- In case of monthly planning, Allevo uses the rate on the first day of the respective period.

Although Allevo respects the period dependency in the exchange rate, a change in the exchange rate during the year can lead to rounding differences. Therefore, please note the following special cases:

- Annual planning: here Allevo uses the conversion rate of the first period to determine the value in the transaction currency for the GP12N call. There, it is internally calculated back to the company code currency, but then at the monthly rate. In total, there will then also be deviations in the posted company code currency. The problem is reduced if the monthly planning in Allevo is used for exchange rate changes during the year.
- Monthly planning: distribution to periods takes place in Allevo in the company code currency. In the case of a linear distribution, the same values are therefore in each month: Allevo then determines the value in transaction currency for each month. In case of exchange rate changes during the year, there is no equal distribution. When planning in GP12N, it is basically the other way around: the starting point is the transaction currency (with linear distribution, if necessary); the company code currency results in the internal, second step. Consequence: this different distribution also leads to different total values (because, for example, it is translated once more in January, once less).

The deviations should not be large in normal cases and cannot be technically circumvented according to the current status.

7.4.4 Custom Specific Ledger Tables (GP12N Adapter)

In individual cases, it may be necessary to align the transfer of plan data to SAP with customer-specific specific ledger tables. The basis for this is a suitable, customer-specific layout for the transaction GP12N: in order to also address this layout in Allevo, a customer-specific ABAP function module must be available (with suitable batch input recording).

Control and call takes place via constant PC_GP12N_ADAPTER (see F1 help).

7.5 Balance sheet planning via the Allevo module "Flexible Planning (FP)"

As of version 3.5.8, balance sheet planning can also be carried out via FP. A suitable Allevo layout must be assigned in order to evaluate the relevant constants such as the entry at PC_READTABLE.

"PC" must also be entered here as OBJECT_TYPE.

The FP constant FD_RECEIVER has no meaning in this case; it makes sense to assign the company code here as well (so the same mapping as for FD_RCV_CCODE).

8 Planning in consideration of the partner Profitcenter

Allevo enables the planning of costs and revenues by specifying a partner (in earlier Allevo versions via row definitions *C* and *D*). In this context, partner always means the "counter" profit center.

The partner is to be entered in the column marked with TimeSet *CY_KEYS* (annual basis) or *CM_KEYS.*

Notice:	SAP does not allow entering the company code for the partner Profitcenter. Therefore, if the partner Profitcenter is assigned to more than one company code, the mere specification of the partner Profitcenter is not unique with regard to the company code.
	However, the company code of the partner cannot be specified explicitly either, since the field is not available in the SAP planning layouts. Instead, the "partner company" is offered. However, this planning feature is not yet supported by Allevo.

8.1 Dynamic readout of the partner relationship

The Allevo dialog corner must be set up on Dynamics (in earlier Allevo versions together with row definition *Z* on the Excel side). Reference is made to the corresponding explanations in the Allevo Excel manual.

In the dynamic area, after reading the reference data, all costs that were booked with partners in the existing reference periods are broken down by cost type and partner Profitcenter and inserted in list form. A previous selection of the partner can be omitted.

Figure 17: Dynamic range (row 106 - 108, before reading out the data)

Provided that the corresponding rows have also been provided with a writing row definition (also *C* or *D*), planning for the specified combination of account and partner Profitcenter can be performed on the same row.

8.2   Inverse posting

The inverse posting makes it possible to directly post the planning of costs/revenues for the specified partner also from the opposite perspective. If the inverse posting is activated, the offsetting entry is automatically executed in the background with reversed signs when planning with the partner Profitcenter. The result is an immediately reconciled planning.

Notice:	In the SAP standard, this offsetting entry is not made automatically. So if Profitcenter A plans a debit from Profitcenter B, the credit from Profitcenter B to Profitcenter A is not made automatically. This has to be done manually. Inverse planning, on the other hand, does this automatically.
	The inverse posting is only possible if the specified partner Profitcenter is unique to the company code. If the partner Profitcenter is assigned to several company codes, no posting can be made.

The activation of the inverse posting is done via the key *ADP* (annual basis) or *ADM* (monthly basis). There a **3** has to be entered on the respective row.

Notice:	When using inverse posting, it makes sense to make an organizational arrangement as to whether the receiver or the sender makes the posting. This prevents the planning from being overwritten by each other.

9 Appendix

9.1 Constants in the Profitcenter planning environment

The following constants affect Profitcenter planning functions.

Constant	Meaning
PC_INT	Read PC data in the CO layout (incl. currency type)
PC_INT_VERSION	PC Planning with the New General Ledger: Actual Data Version
PC_BEKNZ	Activate display separately for debit and credit via debit/credit indicators
PC_NGLA_PLANNING	Activate display separately for debit and credit
PC_READTABLE	Read PC reference data from the New General Ledger
PC_REP_COMP_SEL	Display or select company code in the initial screen
NO_COMPANYCODE	Read reference data without restriction to the company code
IBV_ELIM	Elimination of intercompany sales
LEDGER	Ledger in cost center and Profitcenter planning
NO_ZERO_DELTA	Prevent zero delta planning
PLAN_PC_IN_LOOP	Performance improvement through block by block batch input transfer
MESSAGE_LEVEL	Suppression of error messages, e.g. also helpful if a company code is not assigned to a controlling area.

9.2 Relevant SAP Notes

Here is a selection of relevant SAP notes, especially for Profitcenter planning in the New General Ledger:

1070629   FAQs: New general ledger Migration

826357   Profitcenter Accounting and New General Ledger