



# Allevo & Excel

## User Manual

For Allevo 3.5 and higher

### Efficiently use Excel with Allevo

The advantage of Allevo lies in the combination and integration of two leading software solutions for controlling: SAP and Excel. This manual deals exclusively with the Excel side and should be seen as an extension of the Allevo & SAP manual. It is meant to facilitate the efficient use of Excel and make use of all of its functionalities in combination with Allevo.

With the help of this manual, you will get to know the basic structure of the Allevo Master. You will learn how to plan cost elements, activity types and statistical key figures. You will become confident in defining the layout of your own Allevo Master without hindering the functionality of the program. And last but not least, you will learn helpful tips for how to successfully integrate Excel macros or your own VBA programs.

### Chapters

|           |   |           |
|-----------|---|-----------|
| <b>1</b>  | <b>Introduction.....</b>  | <b>6</b>  |
| <b>2</b>  | <b>Orientation .....</b>  | <b>7</b>  |
| <b>3</b>  | <b>Structure of the Excel file, worksheets.....</b>                       | <b>9</b>  |
| <b>4</b>  | <b>The three Allevo-Excel ground rules .....</b>                          | <b>14</b> |
| <b>5</b>  | <b>Structures within Excel (Corners).....</b>                             | <b>15</b> |
| <b>6</b>  | <b>Global and local object information.....</b>                           | <b>17</b> |
| <b>7</b>  | <b>Main area of the cost element structure.....</b>                       | <b>20</b> |
| <b>8</b>  | <b>Satellite ranges / SatelliteCorner .....</b>                           | <b>46</b> |
| <b>9</b>  | <b>Navigation – the NavigationCorner and the Menu ribbon .....</b>        | <b>54</b> |
| <b>10</b> | <b>Additional settings on the  CUSTOMIZING  sheet.....</b>                | <b>59</b> |
| <b>11</b> | <b>The Dictionary .....</b>   | <b>67</b> |
| <b>12</b> | <b>The totals sheet and the summary sheet .....</b>                       | <b>69</b> |
| <b>13</b> | <b>Allevo Tree: Object selection on the Excel side (only in ABC).....</b> | <b>72</b> |
| <b>14</b> | <b>Allevo MultiObject.....</b>  | <b>74</b> |
| <b>15</b> | <b>Allevo MultiObject Mass (MOM) .....</b>                                | <b>76</b> |
| <b>16</b> | <b>Allevo MultiObject Dynamic (MOD).....</b>                              | <b>81</b> |
| <b>17</b> | <b>MultiObject with variable sheet assignments .....</b>                  | <b>95</b> |
| <b>18</b> | <b>Additional functions in the Allevo Master .....</b>                    | <b>97</b> |

## Table of Contents

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Introduction.....</b>   | <b>6</b>  |
| <b>2</b> | <b>Orientation.....</b>  | <b>7</b>  |
| 2.1      | Using the Allevo Master: Inplace and Allevo Business Client..... | 7         |
| 2.2      | Forms of Allevo.....   | 8         |
| <b>3</b> | <b>Structure of the Excel file, worksheets.....</b>              | <b>9</b>  |
| 3.1      | Standard Allevo Master.....                                      | 9         |
| 3.2      | Worksheet  Total .....   | 10        |
| 3.3      | (Worksheet  SUM ).....   | 10        |
| 3.4      | Worksheet  Start .....   | 10        |
| 3.5      | Main worksheet  Allevo .....                                     | 10        |
| 3.6      | Worksheet  MasterData .....                                      | 10        |
| 3.7      | Worksheet  Customizing .....                                     | 10        |
| 3.8      | Worksheet  Dictionary .....                                      | 11        |
| 3.9      | Worksheet  Infos .....   | 11        |
| 3.10     | Worksheet  Corner .....  | 11        |
| 3.11     | Worksheet  Navigation .....                                      | 12        |
| 3.12     | Worksheet  Satellite .....                                       | 12        |
| 3.13     | Worksheet  Splasher .....  | 12        |
| 3.14     | Worksheet  Structure .....                                       | 12        |
| 3.15     | Worksheet  Style .....   | 12        |
| 3.16     | Worksheet  Changelog .....                                       | 13        |
| <b>4</b> | <b>The three Allevo-Excel ground rules.....</b>                  | <b>14</b> |
| <b>5</b> | <b>Structures within Excel (Corners).....</b>                    | <b>15</b> |
| <b>6</b> | <b>Global and local object information.....</b>                  | <b>17</b> |
| 6.1      | Functional overview.....   | 17        |
| 6.2      | List of available parameters.....                                | 17        |
| <b>7</b> | <b>Main area of the cost element structure.....</b>              | <b>20</b> |
| 7.1      | The controlling elements for column and row definitions.....     | 20        |
| 7.2      | Planning on a yearly or monthly basis.....                       | 21        |
| 7.3      | Yearly basis: key columns.....                                   | 22        |
| 7.4      | Column definitions/TimeSets for transaction data.....            | 24        |
| 7.4.1    | Features and suffix (value key).....                             | 24        |
| 7.4.2    | Default functions for the suffix (value key).....                | 25        |

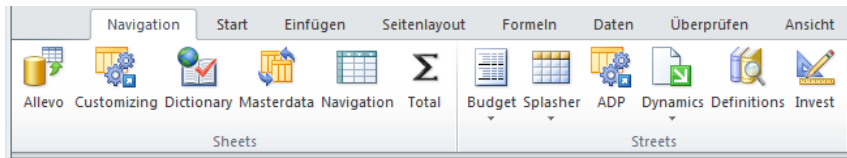
|           |  |           |
|-----------|--|-----------|
| 7.4.3     | Differentiating the “Compact” and Balloon” variants.....                     | 26        |
| 7.5       | Column definitions/TimeSets on a yearly basis: use cases .....               | 26        |
| 7.5.1     | Functional overview (activity independent part).....                         | 26        |
| 7.5.2     | Activity type dependent part (ADP) .....                                     | 28        |
| 7.5.3     | Read and plan prices .....   | 31        |
| 7.5.4     | Distribution key .....   | 31        |
| 7.5.5     | Planning comments.....   | 32        |
| 7.5.6     | Yearly planning using the “Balloon” variant .....                            | 33        |
| 7.6       | Column definitions/Timesets on a monthly basis: Use cases .....              | 33        |
| 7.6.1     | Differences between the “Compact” and “Balloon” variants.....                | 33        |
| 7.6.2     | Key columns for monthly planning .....                                       | 34        |
| 7.6.3     | Setting up monthly planning.....   | 34        |
| 7.6.4     | Balloon variant with monthly blocks one below the other.....                 | 35        |
| 7.7       | Row definitions / AllocationSets .....                                       | 37        |
| 7.7.1     | Operating principle .....  | 37        |
| 7.7.2     | List of Allevo row definitions.....  | 38        |
| 7.7.3     | Dynamic rows (dynamic range).....  | 41        |
| 7.7.4     | Additional functions for reading reference data .....                        | 45        |
| <b>8</b>  | <b>Satellite ranges / SatelliteCorner .....</b>                              | <b>46</b> |
| 8.1       | Control elements of the SatelliteCorner .....                                | 46        |
| 8.2       | Read and write satellite ranges (sequential).....                            | 48        |
| 8.3       | Individually design satellite ranges .....                                   | 49        |
| 8.4       | MatchExistingRow: mode for row accurate reading and writing.....             | 49        |
| 8.5       | Other execution modes for special read and write functions.....              | 50        |
| 8.6       | Number of rows in the data range .....                                       | 51        |
| 8.7       | Formulas and performance in the satellite range (KeyPointer “FORMULA”) ..... | 52        |
| 8.8       | Using “structured tables” with satellites.....                               | 52        |
| <b>9</b>  | <b>Navigation – the NavigationCorner and the Menu ribbon .....</b>           | <b>54</b> |
| 9.1       | Structure of a NavigationCorner .....  | 54        |
| 9.2       | Memory setting of a NavigationCorner.....                                    | 55        |
| 9.3       | Context menu .....   | 56        |
| 9.4       | Toggle button for showing and hiding zero rows (optional) .....              | 57        |
| <b>10</b> | <b>Additional settings on the  CUSTOMIZING  sheet.....</b>                   | <b>59</b> |
| 10.1      | Settings, sheet and workspace protection .....                               | 59        |

|           |  |           |
|-----------|--|-----------|
| 10.2      | Headline – Column headers .....  | 61        |
| 10.3      | UserExits – automatic calling of macros and Callers (views).....           | 62        |
| 10.4      | Global Information .....   | 63        |
| 10.5      | Captions form .....  | 65        |
| 10.6      | Common Flags .....   | 65        |
| 10.7      | Splasher / distribution .....  | 65        |
| <b>11</b> | <b>The Dictionary .....</b>  | <b>67</b> |
| 11.1      | Activating the Dictionary .....  | 67        |
| 11.2      | Functionality of the Dictionary .....                                      | 67        |
| <b>12</b> | <b>The totals sheet and the summary sheet .....</b>                        | <b>69</b> |
| 12.1      | Simple totals sheet (SUM) .....  | 69        |
| 12.2      | Macro based totals sheet (Overview summary sheet) .....                    | 70        |
| <b>13</b> | <b>Allevo Tree: Object selection on the Excel side (only in ABC) .....</b> | <b>72</b> |
| 13.1      | Tree view for object selection in Excel .....                              | 72        |
| 13.2      | Set up the Tree .....  | 73        |
| <b>14</b> | <b>Allevo MultiObject .....</b>  | <b>74</b> |
| 14.1      | Functional overview .....  | 74        |
| 14.2      | Authorizations .....   | 75        |
| 14.3      | Status management .....  | 75        |
| <b>15</b> | <b>Allevo MultiObject Mass (MOM) .....</b>                                 | <b>76</b> |
| 15.1      | General structure of the MOM planning template .....                       | 76        |
| 15.2      | Activating MOM in the settings.....  | 76        |
| 15.3      | Form variants .....  | 77        |
| 15.3.1    | MOM form with static structure .....                                       | 77        |
| 15.3.2    | Dynamic structure via satellite.....                                       | 77        |
| 15.4      | MOM and the use of satellites .....  | 78        |
| 15.5      | MOM in MultiPage-mode (MOMP).....  | 79        |
| 15.6      | MultiObject for PC planning over multiple company codes .....              | 79        |
| 15.7      | Optional: MOM with an explicit entry range (One Page Multi) .....          | 79        |
| <b>16</b> | <b>Allevo MultiObject Dynamic (MOD).....</b>                               | <b>81</b> |
| 16.1      | Functional overview .....  | 81        |
| 16.2      | Examples with the MOD planning input screen .....                          | 81        |
| 16.2.1    | Sorting by object .....  | 81        |
| 16.2.2    | Sorting by cost elements.....  | 82        |

|           |  |           |
|-----------|--|-----------|
| 16.2.3    | Additional sorting options.....                                  | 83        |
| 16.2.4    | Display in BAB-view.....   | 83        |
| 16.3      | Functioning principle for cost elements.....                     | 84        |
| 16.3.1    | Calling the selection input screen .....                         | 84        |
| 16.3.2    | Aggregation according to cost element group.....                 | 85        |
| 16.4      | Functioning principle for statistical key figures.....           | 85        |
| 16.5      | Special functions .....  | 86        |
| 16.5.1    | Calling MOD via the ABC Tree .....                               | 86        |
| 16.5.2    | Additional lists with MOD base information .....                 | 86        |
| 16.5.3    | Breakdown of groups to obtain objects.....                       | 86        |
| 16.5.4    | Use different sorting types / views simultaneously in Excel..... | 87        |
| 16.5.5    | Sheet comments .....   | 87        |
| 16.5.6    | Close planning.....  | 87        |
| 16.6      | Relevant constants and settings in SAP.....                      | 87        |
| 16.7      | Settings on the Excel side.....                                  | 89        |
| 16.7.1    | Setting up the MOD satellite.....                                | 89        |
| 16.7.2    | Configuring StructureBuilder for MOD .....                       | 90        |
| 16.8      | MOD and ProCED .....   | 90        |
| 16.9      | MOD and MultiPage-mode .....                                     | 91        |
| 16.10     | MOD and offline process.....                                     | 92        |
| 16.11     | Optional: MOD expander (show/hide hierarchy levels).....         | 92        |
| 16.12     | Optional: MOD as cost allocation sheet (OAAS) .....              | 92        |
| <b>17</b> | <b>MultiObject with variable sheet assignments .....</b>         | <b>95</b> |
| <b>18</b> | <b>Additional functions in the Allevo Master .....</b>           | <b>97</b> |
| 18.1      | Save any object relevant data in SAP .....                       | 97        |
| 18.2      | Change column properties .....                                   | 97        |
| 18.3      | Customer specific VBA extensions in the Allevo Master .....      | 99        |
| 18.4      | Optional VBA extensions in the Allevo Master .....               | 99        |
| 18.5      | Export as PDF.....   | 99        |

## 1 Introduction

The following chapters describe the use of the most current Allevo Master, which is provided in the Excel format XLSM. XLSM is the VBA-based file type used in Excel versions 2007 and higher. The Master contains functions that were not available in earlier versions of Excel, including the menu ribbon used for navigation in the Allevo Master



**Figure 1.1:** Example of an Allevo menu ribbon

For these reasons, the Allevo Master is no longer compatible with Excel 2003; for such situations, some changes would need to be made to the VBA script. Without the menu ribbon, navigation must be done via Excel objects (forms or controlling elements). However, an XLS Master for earlier Allevo versions (up to 2.9) can be used (at least for Inplace use).

When calling the Allevo Master from SAP, the VBA macros for controlling user inputs and data transfer are automatically executed. The execution of these macros must be authorized in the security settings for the respective Excel program.

Company policies for security settings can vary widely. Where necessary, the Allevo-Master must be adapted to the specific characteristics of these settings, especially if the security guidelines provide for working with certificates (signed Allevo-Master can be provided).

**Note:** To activate the Allevo sheet and work folder protection, see Section 10.1.

## 2 Orientation

### 2.1 Using the Allevo Master: Inplace and Allevo Business Client

There are two formats available for using the Allevo Master:

#### 1. SAP Inplace

In this format, Excel - or more specifically - the Allevo Master, is integrated in the transactions of the SAP system. The user first logs into the system via the SAP login: various Allevo specific transactions are available there, for example, for entering the selection parameters required for planning. When the Allevo Master is called, Excel also opens as an integrated part of the SAP interface (see the detailed description in the Allevo & SAP Manual).

Depending on the Allevo transaction, the Master is called in either “Single, MultiPage or Multi-Object” mode.

#### 2. Allevo Business Client (ABC)

In this format, the Allevo-Master is started by an independent application from the Windows workstation of the planner (e.g., via a desktop icon). A selection window (panel) appears with all the information required for calling an Allevo Master: for example, information regarding the layout and object data. The panel has basically the same functionality as the Allevo transactions in the SAP system. From here, logging into the SAP system is achieved via an RFC connection. From the panel, the planner sees all layouts configured for Allevo in the SAP system and can select them accordingly: Excel is started and opens the selected Master. From there, a stored AddIn takes over the data exchange with the SAP system.

Alternatively, the Allevo Business Client can also be started directly via the Allevo Master (by double-clicking on the file or via a portal). In this way, integration of offline files for data transfer to SAP is also possible.

The Allevo Business Client (referred to here as ABC) is first and foremost an alternative form of data entry. Allevo Customizing (definition of layouts, constant, etc.) is still carried out using the Allevo transactions on the SAP side.

Specific details regarding the ABC format can be found in the Allevo Business Client manual.

Regardless of the use format, the same Allevo Master can be used in both cases. Recommendation:

- Use the Inplace variant if the planners are used to working directly in the SAP system, and special properties, such as jumping to the document, are important. With this type of format, no special software installation is required on the planner’s workstation.
- The ABC variant provides all Excel functionalities, including those that are not supported by the SAP GUI (for example, print preview or status bar functions), depending on the Excel version. This variant also has advantages if direct access from the planning file to other Microsoft environment files is desired.

|              |   |
|--------------|---|
| <b>Note:</b> | In principle, both processes can be operated in parallel. The Allevo Master's program logic automatically recognizes the mode in which it is working. |
|--------------|---|

A user-specific logon is required for both formats. This means that all SAP authorizations and consistency checks are also adhered to when calling via ABC.

## 2.2 Forms of Allevo

The Allevo Master can be used in two different conceptual forms:

- In the “Single” form, one SAP object (profit center, cost center, order, or WBS element) can be processed in the planning form. The structure (for example the cost elements in the P&L structure) is usually stored in a fixed manner in the Master.
- With the “MultiObject” form, all objects are listed in an Excel spreadsheet, whereby different object types (e.g., cost centers and orders) can be mixed together. The structure of this list is usually not rigid, but dynamic. This dynamic response is supported by further Allevo functions, e.g. MOD and ProCED (see detailed descriptions from Chapter 13 onwards).

When working in MultiPage mode, the main worksheet is duplicated as often as objects are selected. Afterwards, the data for all selected objects are available in the same Excel file: this format is preferred for planning with offline files. In MultiPage mode, the main worksheet is usually structured in the same way as in the classic single concept, but the functions of the MOM-mode are also available.

The supplied Allevo-Master template can be used for all of the above-mentioned forms (Single, MultiObject and MultiPage).

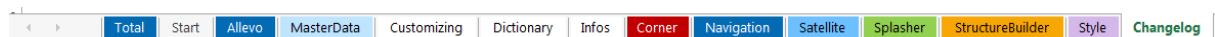


### 3 Structure of the Excel file, worksheets

#### 3.1 Standard Allevo Master

The supplied sample template of the Allevo Master is a standard Master for all SAP object types, as well as for both Allevo forms Single/Multi or MultiObject (MOM for short). In the "delivery state", the Allevo Master (Excel file) contains the following worksheets by default:

- |Total| Totals sheet
- |Start| Help sheet for formulas (empty)
- |Allevo| Main sheet/planning form
- |MasterData| Customer specific parameters, Master data...
- |Customizing| Settings/specifications (UserExits, settings, headlines, etc.)
- |Dictionary| Translations
- |Infos| information on the selected objects of the workbook
- |Corner| Definition tables for technical data
- |Navigation| Control the view and design of the Allevo menu ribbon
- |Satellite| Configure the satellite interface
- |Splasher| Configure the "Splasher" function
- |StructureBuilder| Configure the automated row structure using VBA
- |Style| Configure the automatic cell formatting
- |ChangeLog| Log documenting changes to the file



**Figure 3.1: Worksheets in the Allevo Master**

Differentiation:

- In Single/Multi mode, each SAP object (profit center, cost center, order or WBS element) has its own Excel worksheet as a planning form.
- In the case of MultiPage, the |Allevo| worksheet is copied within the file for each object of the selected object group.
- In Single/Multi mode, you can only work with one object type at a time.
- The MultiObject concept is an exception to this rule. Here you can plan several objects of different types in a list on the main worksheet |Allevo|.

### 3.2 Worksheet |Total|

This overview sheet is mainly used in Allevo's MultiPage mode; all entry objects are listed in columns next to each other to display the values of a single planning scenario (for example, a forecast) for all planning objects. The name of the sheet can be freely selected (for details, see Chapter 12).

### 3.3 (Worksheet |SUM|)

This worksheet does not exist in the supplied template, but can be easily created. This summation sheet is mainly used in Allevo MultiPage mode; it is a copy of the Allevo sheet and adds up the values for all start objects in each value cell.

### 3.4 Worksheet |Start|

This is simply an empty sheet that is required for formulas on the optional summation sheet. If no summation sheet is used, this sheet can be deleted.

### 3.5 Main worksheet |Allevo|

The main worksheet is the actual planning form. From here, the data are transferred to SAP, or the data are transferred from SAP to this sheet (different from the Customizing sheet).

**Important:** A worksheet is recognized by the existence of a range: ROW\_Y, which must be created on the workbook level. In MultiPage mode, or for a MultiObject with several planning worksheets, the range name appears multiple times per sheet. If necessary, it is transferred to the worksheet level by copying the |Allevo| sheet.

|              |  |
|--------------|--|
| <b>Note:</b> | In Master versions prior to 3.0, a fixed name assignment was provided for the worksheet with the planning form, e.g., "MOM_Mass" or "Template". This is no longer required.  |
|              | In earlier Master versions prior to 3.5, the worksheet was recognized by the range name "CC_OBJECT" on the workbook. This is no longer necessary and the information in this range can now be accessed via the  Info  sheet or the local parameter "OBJECT". |

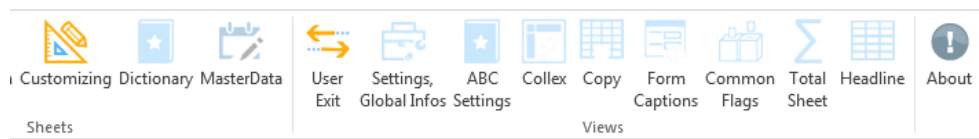
As the first step, you always have to decide whether the Allevo-Master should be used as a Single/Multi- or MultiObject form.

### 3.6 Worksheet |MasterData|

Customer specific tables and formulas, which are required for the main worksheet, are stored here, for example, for generating a drop down menu.

### 3.7 Worksheet |Customizing|

This worksheet is used to make general settings and define specifications or parameters. It is divided into the following elements:



**Figure 3.2: Areas of the worksheet |CUSTOMIZING|**

The individual elements of the worksheet |CUSTOMIZING| will be explained in other sections of this manual:

- User Exit: Section 10.3
- Settings, Global Infos: Sections 10.1 and 10.4
- ABC Settings: see Allevo Business Client manual
- Collex: Tool for data transformation, see separate documentation
- Copy: Tool for copy function, see separate documentation
- Form Captions: See section 10.5 **Fehler! Verweisquelle konnte nicht gefunden werden.**
- Common Flags: See section 10.6
- Headline: See section 10.2
- MultiObject: See section 14
- About: Information on the program version of the current Master

### 3.8 Worksheet |Dictionary|

Allevo supports multi-lingual use of the Excel planning form. To this end, the worksheet |Dictionary| offers the option of translating the menu ribbon and any text entries at a central point in the Master. The required text translations are all managed in a table.

The functions of the dictionary are covered later in this manual (see Chapter 11).

### 3.9 Worksheet |Infos|

The worksheet |Infos| contains all local and global parameters of Allevo. These parameters refer, for example, to object selection, application settings, and the SAP system or logged on user. Since the parameters have been appropriately named, there is no need for an itemized list. The parameters are described in Chapter 6.

### 3.10 Worksheet |Corner|

Here, the structural organization of the Allevo sheet is described and basic settings are defined for configuration with the MasterManager tool. This sheet and its contents are for information purposes only and should not be changed.



### 3.16 Worksheet |Changelog|

The Allevo Master is a central planning element. Changes to the Master can also have unintended effects (for example, incorrect adjustments to formulas for determining plan values). Because of this significance and the potential of such unintended effects, traceability is an important part of the change process: the tabular change log on this worksheet is a design suggestion.

## 4 The three Allevo-Excel ground rules

When configuring the Allevo Master, be sure to consider the following rules:

- (1) In cells containing key values, or those that are relevant for navigation, Excel must **not** return any errors (for example, #NA or #VALUE). These errors can occur if the cell contents are derived from formulas. We recommend that you use the IFERROR and ISERROR functions to detect errors.
- (2) A zero is planned, an empty cell is not. Due to this principle, a complete deletion of data is not possible via Allevo, only a reset to zero. The same applies to formulas: a zero entry is planned, an empty cell created with If (condition; then;"" ) is not.
- (3) Excel cells containing a formula are not overwritten by data from SAP. Allevo assumes that the cell value is recalculated by the formula.

## 5 Structures within Excel (Corners)

As of Allevo 3.5, the central structures in the Excel worksheet |ALLEVO| are set up using corner configurations (aptly called “corners”). Each corner is named according to its function and can exist multiple times within the same worksheet.

Within each corner there are both row-related and column-related assignments. These assignments are made by pairing KeyPointers and Pointers.

To better understand this principle, it may be useful to have a look at a simple example from the Allevo navigation.

The corner *Navigation* is shown here, which consists of a column and row axis. The axes are formed by 2 name ranges, which are structured as follows: <CornerName>Row and <CornerName>Column.

To configure the navigation, KeyPointers are placed first, which assign a row or column within the corner to a specific navigation property.

In the example below, this is the property for displaying a specific column or row under a particular condition (1,2 or \*, Q). The entries that are specifically assigned to the respective KeyPointers are called *Pointers*.

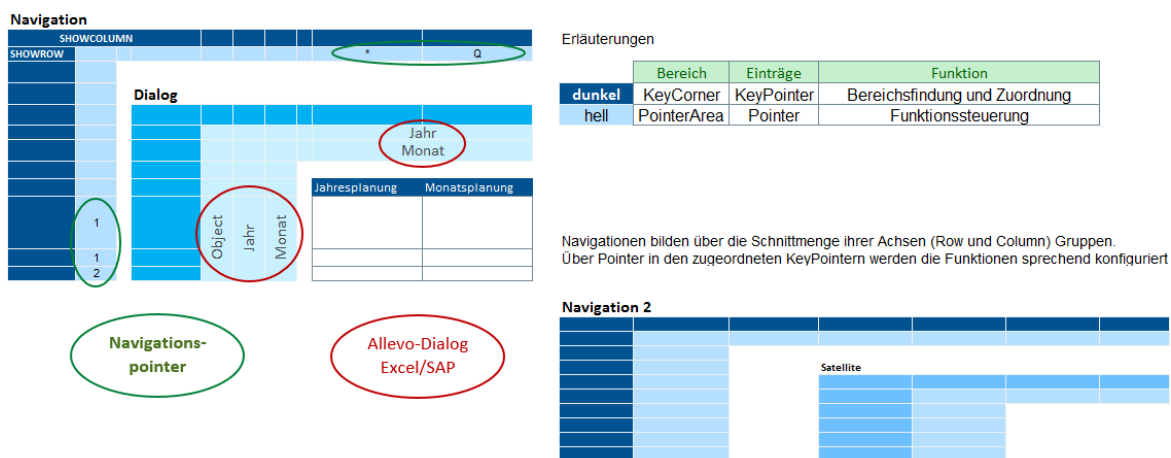


Figure 5.1: General structure of the Allevo Master on the Excel worksheet |ALLEVO|

In Allevo 3.5, the dialog is controlled by additional range names, which will become superfluous in the future. For more information on the dialog, see Chapter 7.

Since each corner always refers to columns and rows, a precise range that is affected by the corner is always configured. Other ranges outside of the corner axes are therefore always independent of the corner and not affected by it.

On the |ALLEVO| sheet there is a corner for the cost element structure - i.e. the direct planning of statistical key figures, activity types and cost elements - as well as a corner for detailed investment planning in satellite 01. The actual number of Streets is therefore dependent on the number of satellites used in the |ALLEVO| worksheet and on the use of additional functions.

In Allevo 3.5, the Allevo dialog with SAP is controlled using control elements that are defined with Excel name ranges.



## 6 Global and local object information

### 6.1 Functional overview

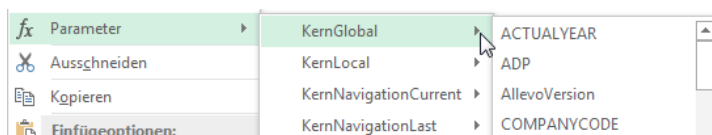
On the worksheet |Infos|, Allevo keeps global and local object information for the following applications:

- Transfer of additional information from SAP to Excel,
- Consistency checks for the case that you are working with a locally saved planning file (offline mode).

**Note:** If you want to continue planning with an offline Allevo file, the local parameters OBJECT, VERSION, YEAR and PERITO are compared with the specifications entered in the Allevo start screen. If the file does not match, there is a conflict. The file will not be opened.

The global object information always refers to the entire workbook and the start object selected when starting planning, whereas the local object information differs for each Excel worksheet.

To access this object information, you can use the Kern parameter functions:



**Figure 6.1:** Insert object information using the parameter function

Of course, it is also possible to use standard Excel formulas to refer to the Infos worksheet. Experience has shown, however, that these are not as intuitive as the parameter formulas.

For use in Allevo MultiObject mode, the reference to the main object of a page is not always sufficient, especially if objects and object types in the respective sheet change. In this case, entries at the row level or for all objects mentioned are of more interest. In this case, constant READ\_ELEMENT\_DATA may be helpful: it provides master data information for all objects in the Allevo Master that are otherwise only available via CC\_CUSTINFOxx. The transfer is done with the help of comment columns.

### 6.2 List of available parameters

Here is a list of the currently available parameters (for both global and local):

| Cell            | Meaning  |
|-----------------|--|
| CONTROLLINGAREA | Controlling area according to the selection in SAP.  |
| OBJECT          | Object number from the SAP internal display (e.g. with leading zeros for cost center and order). In MultiPage mode, this is the object number on the current sheet.  |
| VERSION         | Version for which planning should take place, according to the Allevo settings in SAP (always refers to the year that is entered in the column definition CX_WW on the SAP side). Same contents as GLOBAL_VERSION. |
| PLANYEAR        | Fiscal year according to the Allevo settings in SAP (CX_WW in the column definitions). Same contents as GLOBAL_PLANYEAR.   |

|                |  |
|----------------|--|
| SETCLASS       | Setclass as an identifier of the SAP object type to be processed: for example, "0101" for cost centers, "0103" for orders, "0110" for WBS elements. The cell should be formatted for text.<br>For data transfer to SAP, the object type can also be directly entered (e.g. "KS" as equivalent to "0101" for cost center, see also CC_OBJECTTYPE).  |
| RESPONSIBLE    | The responsible user entered in the SAP master data for the current object (e.g. for the cost center, order, WBS element...).  |
| PERITO         | To period according to the Allevo settings in SAP (CX_RR in the column definitions). Same contents as GLOBAL_PERITO.   |
| TEXT           | Object description (short text from the master record for cost center, order, etc.). For WBS elements, the internal WBS number and level are additionally transferred.   |
| SYSID          | SAP-System used to generate the file   |
| DATE           | Date of file generation  |
| ACTUALYEAR     | Year setting for the actuals reading definition CX_RR  |
| TRANSACTION    | Allevo transaction with which the file was generated   |
| LANGUAGE       | Logon language   |
| LOGSYS         | SAP-System on which the file was generated   |
| GROUP          | Group that is stored in the master record for the starting object  |
| PROFITCENTER   | Profit center that is assigned to the selected object  |
| LSTARS         | Yes/no for reading the dynamic activity relationships  |
| ADP            | Yes/no for activating activity dependent planning  |
| USER           | User that generated the file   |
| LAYOUT         | Allevo layout with which the file was generated  |
| FROM           | For interval selection: From value   |
| TO             | For interval selection: To value   |
| OBJECTGROUP    | For group selection: Group entry   |
| PROJECT        | Only for WBS elements: Assigned project  |
| STATUS         | Allevo planning status for the current object  |
| DESCRIPTION    | Detailed object description (long text from the master record for the object, available for KS, PC, and BP, otherwise the ID of the assigned cost centers)   |
| ALLEVOVERSION  | Allevo version number in SAP   |
| COMPANYCODE    | Company code of the planning object  |
| GROUPNAME      | For group selection: Name of the selected group  |
| FCODE          | Technical function code from SAP   |
| REPRESENTATIVE | Representative from the master record  |
| OBJECTTYPE     | ID of the SAP object type as an alternative to CC_SECTL (Allevo supports KS, BP, OR, PC, PR).<br>See also GLOBAL_OBJECTTYPE with then equivalent entry for the representative object.  |
| COSTCENTER     | Assigned cost center of the object   |
| CUSTINFO1-10   | The fields CC_CUSTINFO01 to CC_CUSTINFO10 can show any additional information for the current object that may be, e.g. stored in the respective SAP master record, but also information such as the name of an assigned 1:n group or an indicator that reporting mode is active (see the detailed F1 documentation for constant MAP_FIELDxx).<br>For use in MO-mode, all information can also be made available at the row level (see constant READ_ELEMENT_DATA). |

|   |   |
|---|---|
| SHEETCOMMENT<br>(Currently still transferrable via name range 'CC_COMMENT') | A long comment for planning can be saved here (sheet comment). This comment is applied generally to the object (cost center, order or WBS element). It can contain more than 255 characters and is saved as an SAP script. The text can be centrally called and edited via the Allevo menu  Satellite tables  (saved in table /KERN/IPPLTEXT) |
| OBJECT2   | Object description (short text from the master record for cost center, order, etc.); For WBS elements, the internal WBS number and level are also transferred.  |
| ISOLANGUAGE   | Logon language of the SAP system with which the data was generated  |
| CURRCONTROLLINGAREA   | Currency of the controlling area (according to CX_WW)   |
| CURROBJECT  | Object currency (according to the entry in the master record of the current object)   |
| TREEVIEW  | Yes/no to activate Tree view  |
| CURRATE   | Current exchange rate from object to controlling area currency (see also note on constant EXCH_RATE_PARAM).   |
| OfflineProcessing   | Information on whether file was generated via offline export  |
| OBJECT1   | Like CC_OBJECT but for external representation (so, without leading zeros).   |

## 7 Main area of the cost element structure

### 7.1 The controlling elements for column and row definitions

The controlling elements for the cost element structure include all Allevo definitions with which the communication between Excel and SAP is controlled.

|              |  |
|--------------|--|
| <b>Note:</b> | The term "cost element structure" is actually somewhat narrow, because of course statistical key figures and activity types can also be included in this structure.                |
|              | When using MultiObject (MOM), the structure also refers to a more or less arbitrary combination of object and cost element (or object and activity type or object and key figure). |

| Y  | Z      | AA    | AB    | AC    | AD    | AE     | AF    | AG     | AH | AI     | AJ | AK    | AL    | AM      | AN       | AO          | AP          | AQ       | AR       | AS      | AT | AU |
|----|--------|-------|-------|-------|-------|--------|-------|--------|----|--------|----|-------|-------|---------|----------|-------------|-------------|----------|----------|---------|----|----|
| 24 | Dialog |       |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 25 | MO_AO  | AOT   | YW    | YR    | YR2   | MW     | MR    | MR2    |    | CY_KEY |    |       |       |         |          |             |             |          |          |         |    |    |
| 26 | YEAR   | MO_AO | AOT   | MO_YW | MO_YR | MO_YR2 |       |        |    | CY_KEY |    | CY_AD | CY_AD | CY_KEYS | CY_KEYSA | CY_KEYRTYPE | CY_KEYR     | CY_KEYRA |          |         |    |    |
| 27 | YEAR   | MO_AO | AOT   | MO_YW | MO_YR | MO_YR2 |       |        |    | CY_KEY |    | CY_AD |       | CY_KEYS | CY_KEYSA | CY_KEYRTYPE | CY_KEYR     | CY_KEYRA | CY_R3_Q  | CY_R3_V |    |    |
| 28 |        |       |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 29 | MON    | MO_AO | AOTYP |       |       | MO_MW  | MO_MR | MO_MR2 |    | CM_KEY |    |       |       | CM_ADF  | CM_KEYS  | CM_KEYSA    | CM_KEYRTYPE | CM_KEYR  | CM_KEYRA |         |    |    |
| 30 | MON    | MO_AO | AOTYP |       |       | MO_MW  | MO_MR | MO_MR2 |    | CM_KEY |    |       |       | CM_ADF  | CM_KEYS  | CM_KEYSA    | CM_KEYRTYPE | CM_KEYR  | CM_KEYRA |         |    |    |
| 31 | MON    | MO_AO | AOTYP |       |       | MO_MW  | MO_MR | MO_MR2 |    | CM_KEY |    |       |       | CM_ADF  | CM_KEYS  | CM_KEYSA    | CM_KEYRTYPE | CM_KEYR  | CM_KEYRA |         |    |    |
| 32 |        |       |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 33 |        |       |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 34 |        |       |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 35 |        |       |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 36 |        |       |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 37 | 1000   | KS    | Q     | Q     |       | Q      | Q     |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 38 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 39 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 40 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 41 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 42 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 43 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 44 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 45 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 46 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 47 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 48 | 1000   | KS    | A     | A     | I     | A      | A     | I      |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 49 | 1000   | KS    | A     | A     | I     | A      | A     | I      |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 50 | 1000   | KS    | A     | A     | I     | A      | A     | I      |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 51 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 52 | 1000   | KS    | A     | A     | I     | A      | A     | I      |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 53 | 1000   | KS    | A     | A     | I     | A      | A     | I      |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 54 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |
| 55 | 1000   | KS    |       |       |       |        |       |        |    |        |    |       |       |         |          |             |             |          |          |         |    |    |

Figure 7.1: Controlling elements for the cost element structure

- In cell AB37, a three-column definition range "ROW\_Y" begins, which designates the sheet as an Allevo sheet. This sheet is used to transfer annual values between SAP and Excel.
- Next to the year range, there is a three column month range "ROW\_M", which begins at cell AE37.
- The three row Excel range Z26:AU28 (and, if necessary, further columns to the right) contains the Allevo column definitions for annual planning. The entire range here is defined with the Excel range name "COL\_Y".
- The two column Excel range Z37:AA55 contains the object list along with "Setclass / Object Type" for using the Allevo Master as a MultiObject form. Both columns can be positioned independently; this information is determined by creating intersections using the KeyPointers MO\_AO or MO\_AOTYP with the entire rows of the ROW\_Y/ROW\_M range (which are used for the information SETCLASS and OBJECTTYPE).

**Tip:** Use the keyboard combination Ctrl+G (go to) to go to the named ranges.

Allevo uses definitions via Excel ranges, which can be named using the standard Excel approach (Excel Menu |Insert – Name – Define| or the keyboard combination Ctrl+F3). These definitions must be contained on the |ALLEVO| worksheet. If a definition is missing here, an error message will appear when calling the Allevo Master in SAP.

The Excel range names are permanently fixed to the selected cell range. If columns or rows are entered or deleted, then the ranges are changed or moved accordingly. Through the stability of the connection to a cell, you can achieve great freedom during individual design of the layout.

**Note:** The systematics of the Allevo row and column definitions are closely related to the Allevo settings on the SAP side, found in the administration menu |Settings|, folder |Column/row definitions|. Please refer to the corresponding explanations in the Allevo & SAP manual.

Descriptions (e.g. for cost elements) are usually permanently stored in the Master. In some cases, however, it may be useful to transfer these descriptions directly from SAP (language dependent). In this case, the constant READ\_ELEMENT\_TEXTS can be used.

## 7.2 Planning on a yearly or monthly basis

The column definitions for planning on either a monthly or a yearly basis are different:

- COL\_Y Excel range name for planning on a yearly basis
- COL\_M Excel range name for planning on a monthly basis

For planning on a yearly basis, you can differentiate between activity independent and activity dependent planning.

|                                    |                                     | Jahresplanung                      |      |      | Monatsplanung 1. Jahr        |     |       |     |     | Monatsplanung 2. Jahr |     |     |     |     |      |     |     |     |     |  |
|------------------------------------|-------------------------------------|------------------------------------|------|------|------------------------------|-----|-------|-----|-----|-----------------------|-----|-----|-----|-----|------|-----|-----|-----|-----|--|
|                                    |                                     | leistungsarten- <b>un</b> abhängig |      |      | <b>-abhängig</b> , Plan 2010 |     |       |     |     | 2010                  |     |     |     |     | 2011 |     |     |     |     |  |
|                                    |                                     | Ist                                | Ist  | Plan | für Leistungs- ..            |     |       |     |     | Jan                   | Feb | ... | Nov | Dec | Jan  | Feb | ... | Nov | Dec |  |
|                                    |                                     | 2008                               | 2009 | 2010 | ... art 1                    | ... | art 2 | Jan | Feb | ...                   | Nov | Dec | Jan | Feb | ...  | Nov | Dec |     |     |  |
| <b>Zeilendefinition Jahr ROW_Y</b> | <b>Zeilendefinition Monat ROW_M</b> | <b>Kennzahl</b>                    |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | Mitarbeiter                        |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | Fläche                             |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | <b>Leistungen</b>                  |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | Stunden                            |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | Liter                              |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | <b>Kostenarten</b>                 |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | Personal                           |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | Material                           |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
|                                    |                                     | Beratung                           |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |
| ILV                                |                                     |                                    |      |      |                              |     |       |     |     |                       |     |     |     |     |      |     |     |     |     |  |

Figure 7.2: Section from yearly planning with Excel ranges for column and row definitions

The column structure in Allevo is defined by the Excel ranges COL\_Y and COL\_M. The ranges should therefore be created for all Excel columns that are to be used in yearly and monthly planning.

In Allevo Standard, COL\_Y and COL\_M are applied to the same columns. The Allevo ranges ROW\_Y and ROW\_M are applied to the same rows.

The combinations COL\_Y/ROW\_Y and COL\_M/ROW\_M can also be arranged in a staggered way. If you wish to do this, please contact us.

### 7.3 Yearly basis: key columns

The column organization in the Allevo Master orients itself according to the customary SAP report structure, consisting of key columns for cost elements, for example, as well as columns for the transaction data (the observed periods and value types).

Figure 7.3: Yearly planning and the corresponding Excel range name (red arrow)

The range COL\_Y, where you enter Allevo column definitions always consists of two or optionally three connected rows. The first row contains the column definitions for planning, the second row contains the definitions for reading data.

- Key column in Figure 7.3: Columns Q-X
- Transaction data in Figure 7.3: Columns AA-...

In Allevo 3.5, the third row for yearly planning is not yet filled with individual pointers, however the third column makes it possible to read and plan without entering the dimension ID (\_V, \_Q, etc.). In these cases, the Allevo row definitions determine whether the transferred values should be interpreted as a quantity or as an amount.

The key columns define in which Excel columns the data for cost elements (or statistical key figures or activity types), possible partner objects (sender cost center, sender activity type) or SAP distribution keys are expected. Allevo definitions for key columns should always be in both rows, that is, in the rows for both the read and write definitions:

| Allevo column definition for: |          | Column function   |
|-------------------------------|----------|---|
| Reading                       | Planning |   |
| MO_AO                         | MO_AO    | MultiObject: Object key (e.g. cost center).<br>Alternatively, a group can also be entered here (for example, cost center group after selection in the Tree). Allevo then reads the data of all objects contained within in total, similar to 1:n planning. Reading groups is mainly intended for reporting functions (for details, see constant MULTI_WITH_GROUPS). |

| Allevo column definition for: |             | Column function  |
|-------------------------------|-------------|--|
| Reading                       | Planning    |  |
|                               |             | With this column, a suitable Excel cell format must be used (e.g. "Text" instead of "Standard") so that the object numbers are understood correctly when transferred to SAP (may be crucial, for example, with numeric object IDs).  |
| MO_AOTYP                      | MO_AOTYP    | MultiObject: Object type or setclass of the object.<br>Both keys are also possible (i.e. KS or 0101 for cost centers; OR or 0103; PR or 0110; PC or 0106).<br>See the parameter UseMOMObjectType in the Allevo settings.   |
| CY_KEY                        | CY_KEY      | SAP Key for cost element, activity type, statistical key figure or group (e.g. cost element group)   |
| CY_DISKEY                     | CY_DISKEY   | SAP distribution key has to exist on the SAP side (not applicable as of version 3.0). Please set up new definitions (see Section 7.5.4)  |
| CY_KEYS                       | CY_KEYS     | SAP key of the sender cost center for activity allocation  |
| CY_KEYSTYPE                   | CY_KEYSTYPE | SAP sender object type (dynamic range)   |
| CY_KEYSA                      | CY_KEYSA    | SAP key of the sender activity type for activity allocation  |
| CY_KEYR                       | CY_KEYR     | 1. Receiver object for crediting from activity allocation.<br>When using constant SPLIT_PAROB only the SAP key for the receiver cost center (dynamic range X)<br>2. Special case for "Quantities for primary costs via row definition B" for transferring an individual quantity unit (see Allevo & SAP manual)<br>3. Transfer of special features when reading and writing, e.g., for the functional area (see constant READ_WITH_KEYR) |
| CY_KEYRTYPE                   | CY_KEYRTYPE | SAP receiving object type (dynamic range X)  |
| CY_KEYRA                      | CY_KEYRA    | SAP key of the receiver activity type for crediting from activity allocation (see constant SPLIT_PAROB, dynamic range X)   |
| CY_ADP                        | CY_ADP      | Controls whether the cost element or statistical key figure under CY_KEY is planned as activity type dependent.<br>An entry of <b>1</b> or <b>2</b> activates activity dependent planning via Excel (see Section 7.5.2)<br>An entry of <b>3</b> or <b>4</b> makes it so that the activity dependent planning is controlled via the defined set of rules (see Section 7.7.2)  |

For using MultiObject via Excel assignments, the following column definitions play a role:

| Allevo column definitions for: |          | Column function                      |
|--------------------------------|----------|--------------------------------------|
| Reading                        | Planning |                                      |
| MO_YW/MO_MW                    |          | Plan row definition for year/month   |
| MO_YR1/MO_MR1                  |          | Read row definition for year/month 1 |
| MO_YR2/MO_MR2                  |          | Read row definition for year/month 2 |
| MO_Var1                        | MO_Var1  |                                      |

## 7.4 Column definitions/TimeSets for transaction data

### 7.4.1 Features and suffix (value key)

The columns for the transaction data refer to the corresponding column groups (column definitions) of the Allevo settings in the SAP system. You specify the years/periods, versions and value types to which the data shown in Excel refers (both for reading and planning). The basic structure and meaning of the abbreviations used are described in the Allevo & SAP manual.

|              |  |
|--------------|--|
| <b>Note:</b> | <p>The term “column definition” is derived from the fact that it results in the properties of the Excel column in whose header the ID code is entered. In newer Allevo versions, this ID code can also be entered in a row (e.g., to view the monthly actuals and plan values in two adjacent rows). For this reason, the term “TimeSet” is also sometimes used.</p> <p>Common ID codes include “CY_RW, CX_WW, CM_R1”. In earlier program versions, there were some restrictions on the allocation of the names (for example, structures according to the pattern CY_Rn and CY_Rn for reading). Today, there are other keys such as CY_RAA or CY_RP1, for using ID codes that are more descriptive.</p> <p>The validity of a key is always checked on the ABAP side.</p> |
|--------------|--|

On the Excel side, a suffix (additional ending) for the column group describes which quantities/amounts or keys are to be addressed using a column group defined in SAP. The suffix defines the data type. Since the introduction of the three-row year header (COL\_Y), it has been possible to dispense with this suffix in order to leave the decision on transferring the dimension (quantity/amounts) to the row definition alone. In this way, you can use the same column to read and plan quantities for statistical key figures and amounts for cost elements, for example. See also Section 7.4.2.

|             |   |
|-------------|---|
| <b>Note</b> | <p>On the Excel side, CY_RW_V is used to read an amount via the group CY_RW; CY_RW_Q would read a quantity (e.g. for a row with key figures).</p> |
|-------------|---|

Some suffixes only make sense for columns with entries for years (e.g. distribution key). The following are currently available:

| Suffix | Meaning  | Type     | L Read<br>P Plan<br>X Both | J Year<br>M Month<br>X Both |  |
|--------|--|----------|----------------------------|-----------------------------|--|
| _V     | Value  | Value    | X                          | X                           |  |
| _V2    | Value 2 (for a 2 <sup>nd</sup> currency)             | Value    | X                          | X                           |  |
| _VF    | Value fix (activity type dependent planning)         | Value    | X                          | X                           |  |
| _Q     | Quantity   | Quantity | X                          | X                           |  |
| _QF    | Quantity fix (activity type dependent planning)      | Quantity | X                          | X                           |  |
| _QV    | Quantity variable (activity type dependent planning) | Quantity | X                          | X                           |  |
| _QM    | Quantity maximal (only for statistical key figures)  | Quantity | L                          | L                           |  |



| Suffix | Meaning  | Type     | L Read<br>P Plan<br>X Both | J Year<br>M Month<br>X Both |  |
|--------|--|----------|----------------------------|-----------------------------|--|
| _PF    | Price fix  | Price    | X                          | L                           |  |
| _PV    | Price variable   | Price    | X                          | L                           |  |
| _PU    | Price unit (Price per xx units)                                    | (Text)   | L                          | L                           |  |
| _N     | Notes  | Text     | X                          | J                           |  |
| _DKF   | Distribution key (fix)   | Text     | X                          | J                           |  |
| _DKV   | Distribution key (variable)  | Text     | X                          | J                           |  |
| _C     | Capacity (activity types)  | Quantity | L                          | L                           |  |
| _D     | Activity scheduled   | Quantity | L                          | L                           |  |
| _O     | Output (activity types)<br>(reserved for future developments)      | Quantity | -                          | -                           |  |
| _EN    | Equivalence (activity types)<br>(reserved for future developments) | (Text)   | -                          | -                           |  |

As you can see, the first character in the suffix always creates a basic differentiation of content, for example, V = Value, Q = Quantity, D = Distribution key.

|             |  |
|-------------|--|
| <b>Note</b> | The list above is up to date as of Allevo 3.5. For restrictions in previous versions, please refer to an earlier manual. |
|-------------|--|

### 7.4.2 Default functions for the suffix (value key)

It is not always advisable to separate quantity and value columns. An unnecessarily large number of columns would result, especially when planning using periods. For this reason, Allevo makes it possible to work with default functions for the suffix.

|                 |  |
|-----------------|--|
| <b>Example:</b> | Without any suffix, the SAP side decides which content is to be read, for example, quantity for the statistical key figure and activity type or value for primary costs, based on the respective row definition. On the SAP side, the respective suffix is derived (for example " for row definition A for primary costs). |
|-----------------|--|

For differentiating between fixed and variable portions, Allevo uses a pseudo-suffix “\_FIX” and “\_VAR”. The last three columns in the following table describe which suffix the SAP side uses in each case.

| Area               | Row definition | Default<br>(no Suffix) | _VAR | _FIX |
|--------------------|----------------|------------------------|------|------|
| CO-primary costs   | A, B, I ,J     | _V Value               | _VV  | _VF  |
| CO-secondary costs | C, D           | _Q Quantity            | _QV  | _QF  |

| Area                             | Row definition                          | Default (no Suffix) | _VAR | _FIX |
|----------------------------------|---|---------------------|------|------|
| Statistical key figures          | Q, R, S                                 | _Q Quantity         | _QV  | _QF  |
| Activity types                   | M                                       | _Q Quantity         | _QV  | _QF  |
| Capacity of activity types       | N                                       | _C Capacity         |      |      |
| Profit center without quantities | A, C, E, G, I<br>P (for PC integration) | _V Value            | _VV  | _VF  |
| Profit center with quantities    | B, D, F, H, J                           | _Q Quantity         | _QV  | _QF  |

For the different properties of the row definitions mentioned above, see the explanations below.

### 7.4.3 Differentiating the “Compact” and Balloon” variants

In order to be able to use the column definitions presented here, the corresponding ID code must be entered in one of the following ranges of the Allevo Master:

- COL\_Y Excel range names for planning on a yearly basis
- COL\_M Excel range name for planning on a monthly basis

These ranges can contain 2 or 3 rows: by now, the “Balloon” variant with 3 rows is common, which has been available for monthly planning since Allevo 3.0 and for yearly planning since Allevo 3.4. The data types listed above as suffixes of the column definitions apply to this variant.

Differences between the two variants are most visible in monthly planning:

- The "compact" procedure can plan for 12 or 24 months exact-to-the-month. However, it does not differentiate between quantity or value columns, so that primary cost columns always contain values, and activity allocation always contains quantities.
- The "Balloon" variant can plan any number of periods in any number of years. You can differentiate between quantity and value columns by using the suffixes (\_Q/\_V) as described above.

Within yearly planning, the advantages of the "Balloon" variant are particularly evident in the extended range of data types that can be used as suffixes for column definitions. The data types listed in the section above also only apply to this variant.

The older compact procedure is only partially described in the manual here (for details please refer to earlier manuals).

## 7.5 Column definitions/TimeSets on a yearly basis: use cases

### 7.5.1 Functional overview (activity independent part)

The columns for the transaction data refer to the corresponding column definitions from the Allevo settings. You specify the years/periods, versions and value types to which the data shown in Excel refers (both for reading and planning).

|              |  |
|--------------|--|
| <b>Note:</b> | The properties of the column definitions are assigned on the SAP side in the Allevo administration menu  Properties , folder  Column/row definitions . |
|--------------|--|

See the Allevo & SAP manual for further information.

The column definitions for yearly planning must be entered in the Excel range COL\_Y. The range usually consists of three rows (“Balloon” variant):

- The first row contains the column definition for planning
- The second row contains the column definition for reading
- The third row is reserved for future enhancements; it has no function at this time (it is used internally only to distinguish it from the previous, two-line compact procedure)

The following table describes some common ID codes for column definitions:

| Allevo column definition for:                                |  | Column function   |
|--|--|---|
| Reading  | Planning   |   |
| CX_RR_<extension><br>(z. B. CX_RR_V)                         |  | Allevo column definition for reading actuals data from the current year.<br>CX_RR can also be used as the definition for exact-to-the-month reading of actuals data for year 1.   |
| CY_RW_<extension><br>(z. B. CY_RW_V)                         |  | Allevo column definitions for the reading of plan data for the main planning year. The settings correspond to CX_WW. As of Allevo 3.5, it is possible to do away with this version in many cases, since CX_WW can also be used directly as a reading reference.   |
| CY_R1_<extension> –<br>CY_R99_<extension><br>(z. B. CY_R1_V) |  | Further possible column definitions for reading data.   |
|  | CX_WW_<extension><br>(z. B. CX_WW_V)                         | Allevo column definition for the main planning year (writing). The settings for CX_WW is decisive for the status in status management.<br>CX_WW can also be used as the only definition for activity type dependent planning (in the direct approach).<br>CX_WW is also used as a writing definition for the exact-to-the-month planning of year 1. |
|  | CY_W1_<extension> –<br>CY_W99_<extension><br>(z. B. CY_W1_V) | Other possible column definitions for writing data. You can only use activity-type-dependent planning for these definitions if you plan using the set of rules.   |

In addition to the column definitions CX\_RR, CY\_RW (reading) and CX\_WW (writing), any other column definitions can also be used, e.g. CY\_R1 and CY\_R2 for reading columns and CY\_W1 and CY\_W2 for writing columns.

**Note:** The entries in the table above correspond to an Allevo standard for the names of the column definitions, since in earlier program versions the names had to be assigned with such restrictions (e.g. structure according to the pattern CY\_Rn\_Q and CY\_Rn\_V for reading).

Today, however, keys such as CY\_RAA\_Q or CY\_RP1\_V are also possible, for using ID codes that are more descriptive. The validity of a key is always checked on the ABAP side.

Further information for use in the Allevo Master:

- All column definitions that read or transfer data from SAP can be fit with a suffix on the Excel side, which characterizes the data type of the column (a list of possible suffixes can be found in Section 7.4 above).
- The order of the Allevo column definitions should not matter as long as they are entered in the cells in the range COL\_Y.
- Reading and planning definitions can be in one column, since they are addressed by different actions (read reference data, transfer plan data). As of Allevo 3.5, the same columns can be used for both reading and writing. This read/write direction can be configured on the SAP side and restricted if necessary.
- Any number of additional columns can be inserted, e.g. for calculation operations or deviation analyses, which do not have to be provided with Allevo column definitions.

### 7.5.2 Activity type dependent part (ADP)

Activity type dependent planning is only relevant for cost centers. The following information only applies if you use the "Direct Procedure" via Excel for activity type dependent planning (ADP). When using the "rule planning" via an Allevo set of rules on the SAP side, this part can be omitted in the Allevo Master.

For a more detailed description of the Direct Procedure and Rule Planning, see the Allevo & SAP Manual.

The Allevo Master also uses a column system to map the activity type dependencies. This is an extension of yearly planning and supplements the standard part of activity type independent planning with additional columns in which the individual activity types of a cost center are addressed differently for planning. If the ADP is to be carried out on a monthly basis using the "direct procedure", please refer to Section 7.6.

You can plan up to 20 activity types for each cost center, in an activity type dependent manner. From a technical point of view, the costs are not assigned to the cost center (KS object), but more precisely to an activity type of the cost center (KL object). To be able to make this assignment for each activity type, there are up to six columns (CYA\_1, CYA\_2, CYA\_3, each with a distinction in \_VF= "Value Fix" and \_VV= "Value Variable") in the standard Master for each activity type. Using these columns, you can plan quantities and values, differentiated by fixed and variable portions, with fixed and variable distribution keys. In Excel, up to 20 differentiations per activity type (CYA\_1 to CYA\_20) are possible, since CYA\_0 always forms the activity type independent part. There is no upper limit to the number of dependent activity types in the Allevo SAP rule framework.

The implementation on the Excel side is explained briefly here: For the ADP, the Excel range COL\_Y on the |ALLEVO| sheet is extended by additional columns. Activity type dependent planning has its own Allevo column definitions, which are to be entered in the COL\_Y range. To assign the columns to an activity type, an additional Excel range name is inserted for each activity type. This Excel range name groups together the corresponding columns like a bracket. The result is that the columns belonging to an activity type are displayed in a column group.

|       |                                    |          |          |            |          |          |
|-------|------------------------------------|----------|----------|------------|----------|----------|
| CYA_1 | =WENN(ANZAHL2(MainActivityType)>=E |          |          |            |          |          |
|       | BQ                                 | BR       | BS       | BT         | BU       | BV       |
| 24    |                                    |          |          |            |          |          |
| 25    | CYA_DKF_1                          | CYA_QF_1 | CYA_VF_1 | CYA_DKV_1  | CYA_QV_1 | CYA_VV_1 |
| 26    | CYA_DKF_1                          | CYA_QF_1 | CYA_VF_1 | CYA_DKV_1  | CYA_QV_1 | CYA_VV_1 |
| 27    | 1                                  |          |          |            |          |          |
| 31    |                                    |          |          |            |          |          |
| 32    | 1520                               |          |          |            |          |          |
| 33    | DisKey fix                         | Q fix    | V fix    | DisKey var | Q var    | V var    |

Figure 7.4: Structure of a column group per activity type (complete, example activity type 1410)

In the standard Master, the ADP zone is placed between the yearly and monthly zone.

**Important:** The ID codes and range names used below already refer to definitions that have been common up to Allevo version 3.4. As of Allevo version 3.4, the so-called balloon variant with a three-row COL\_Y range is also available for yearly data. In this context, some range names have also changed (see Section 7.5.6).

These Excel range names are called (up to Allevo 3.3):

- CYA\_0 for activity type independent quantities and values,
- CYA\_1 for the quantities/values of the 1st activity type of a cost center,
- ...
- CYA\_20 for the quantities/values for the 20th activity type of a cost center

In the cell ranges from CYA\_1 to CYA\_20 in Excel, the actual SAP keys for the activity types of a cost center should be entered (directly or derived from a formula, such as Service Hours in the example in Figure 7.5). Of course, there only need to be so many cell ranges named with CYA\_1 to CYA\_20 as the maximum number of possible activity types per cost center.

- Within the framework of activity type dependent planning, it is also possible to plan independent values and quantities. For this purpose, there is the independent column block (CYA\_0).
- Figure 7.5 illustrates that activity type dependent planning is based on the row structure of activity type independent planning. It is therefore based on the same range ROW\_Y for controlling the Allevo row definition (see Section 7.6).
- Planning is carried out according to SAP specifications, i.e. the value columns are used for primary costs and the quantity columns for activity inputs.
- To plan a cost input on an activity type of the cost center, the KL object must exist for the relevant fiscal year and the version, that is, at least one plan price has to exist for the activity type.

**Notes:** In the Allevo Master, not all column definitions of a column group must be used. For example, you can omit the columns for fixed and variable quantities in pure value planning.

There are no variable components for the independent part (CYA\_0). Therefore, the Allevo column definitions are sufficient for fixed quantities and values. The independent part is only necessary if you want to plan independently and dependently simultaneously for a single cost element of the same cost center.

Main area of the cost element structure

- Please note that the range names CYA\_0, CYA\_1, etc. must always cover all columns that contain the Allevo column definition for the respective activity type.
- If you want to write comments for each activity type in activity type dependent planning, you must use separate column definitions for each activity type.
- In individual cases, it may be useful to store the activity types in satellite tables and fill the CYA\_X range from there. This makes it possible to manage activity types dependent on cost centers.
  - In this case, the constant READ\_LSTARTABLE must also be set to ensure that the correct sequence is followed when transferring data between Excel and SAP.

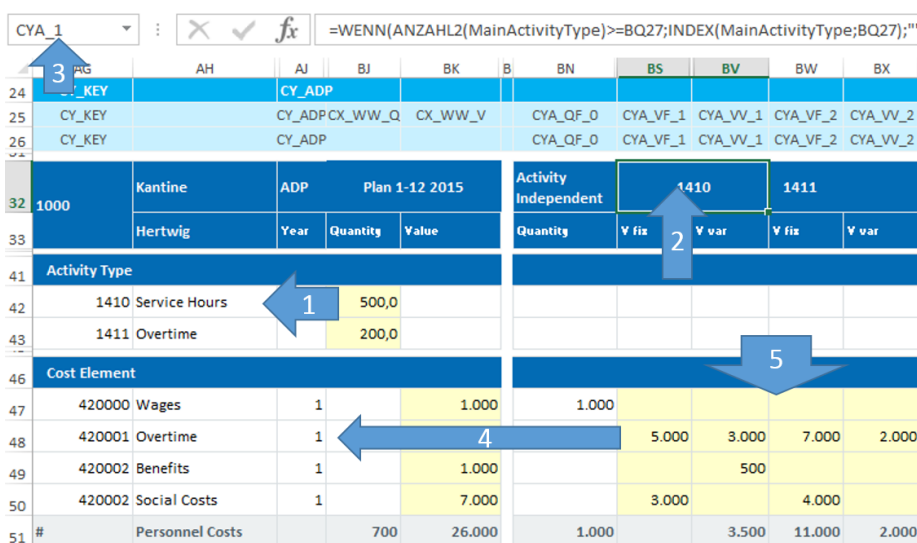


Figure 7.5: structure of the activity type dependent part in yearly planning for 2 activity types

The activity type (1) from quantity planning (KP26) repeats itself as a column (2) in activity type dependent cost planning. The cell of this column in which the activity type is located receives an Excel range name (3). The cost elements are activated in the AM column (column definition CY\_ADP) for activity type dependent planning (4). The cost element is then planned according to activity type in the relevant column ((5) columns AS - AV). The columns for activity type dependent quantity planning (activity inputs) are missing in the figure above.

| Complete access key | Column configuration      | Range name for respective activity type |
|---------------------|---------------------------|---|
| CYA_0_DKF           | Distribution key          |   |
| CYA_0_QF            | Quantity fix              |   |
| CYA_0_VF            | Value fix                 |   |
| CYA_1_DKF           | Distribution key fix      |   |
| CYA_1_QF            | Quantity fix              |   |
| CYA_1_VF            | Value fix                 |   |
| CYA_1_DKV           | Distribution key variable |   |

|  |                   |  |
|--|-------------------|--|
| CYA_1_QV   | Quantity variable |  |
| CYA_1_VV   | Value variable    |  |
| ...The following pattern applies for other activity types... |                   |  |
| CYA_n_dF   |                   |  |
| CYA_n_dV   |                   |  |

The following constants control further functions of activity type dependent planning:

- **BUTTON\_LSVAR** Button for displaying the rules from the rule framework
- **LSTAR\_VARIATOR** Activate rules for activity type dependent planning
- **PLAN\_AEQUZIFF** Transfer the equivalence number for the activity type from Excel to SAP
- **TARIF\_0\_PLAN** Activity planning for KL objects without a price
- **TARIF\_KZ** Price number and price unit
- **READ\_ADP** Rule planning based on reference data
- **READ\_LSTARTABLE** Sequence for reading activity types via satellite
- **SAVE\_ACTDEP** Save entries for CY\_ADP/CM\_ADP
- **SPLIT\_PAROB** Activity dependent planning with split KL object
- **USE\_DYN\_ACTDEP** Dynamic range: display receiver for activity inputs

Activity type dependent planning over several years: If an activity type is entered via CY\_KEYRA, Allevo can also read and plan in an activity type dependent manner without specifying CY\_ADP (this should then be done separately for fixed and variable portions, the total value is always read for months). Alternatively, a row can also work with an empty entry in CY\_KEYRA to read the activity type independent portions (this should be done using a separate row definition).

### 7.5.3 Read and plan prices

To read prices, the corresponding option must be set in the Allevo column definition (see also constant TARIF\_KZ).

By default, prices for secondary costs are only read if values have been posted beforehand. Using the constant READ\_SEC\_TARIF ensures that the prices are always read as well (that is, also for new planning).

In connection with the planning of activities on cost centers (SAP transaction KP26), prices can also be planned. For this purpose, the writing column definitions (CX\_WW, CY\_W1, etc.) can be provided with the corresponding extensions PV, PF, PU (e. g. CX\_WW\_PV, CX\_WW\_PF, CX\_WW\_PU) and inserted as a column in the planning form. All three column definitions must exist (see Section 7.5.1).

|              |   |
|--------------|---|
| <b>Note:</b> | Prices are only planned if an activity quantity is also planned. If no price unit is given, the value 1 is used as the default. |
|              | Prices are only planned on a yearly basis.  |

### 7.5.4 Distribution key

As of Allevo 3.0, it has been possible to define the distribution keys not only generally at the cost element level, but also depending on the column definition. In the case of multi-year planning, you can

therefore plan each year using a separate distribution key. The system differentiates between distribution keys for "fixed" (\_DKF) and "variable" (\_DKV).

Constants with functions relating to the distribution key:

- **PLAN\_DISKEY** This constant makes it possible to transfer a distribution key (fixed and variable) from the Allevo Master when using rule planning.
- **PLAN\_DK\_FIXVAR** Using this constant allows the fixed distribution key for cost centers to be transferred to the variable distribution key.
- **READ\_DISKEY** Constant for reading the last planned distribution key
- **READ\_DISKEY\_ALL** Constant for reading the last planned distribution key

### 7.5.5 Planning comments

The row comments function can be activated for each column definition "to be planned" (suffix "\_N"). This makes it possible to save and read comments for each fiscal year or plan version, even in the case of multi-year planning.

The row comments are stored for the cost element, activity type, or statistical key figure. These comments are transferred to SAP with each row definition in the first column of the ROW\_Y range (Year write). In order to transfer row comments to SAP for monthly planning, you can use the row key "T" (only comments) in the year write/read range.

|              |   |
|--------------|---|
| <b>Note:</b> | The entry <b>T</b> is also suitable for entering a comment at the level of a totals row. In the key column CY_KEY, you can make an entry of your choice (in capital letters or numbers) with a maximum length of 10 characters. |
|--------------|---|

When reading the comments, make sure that the "reading" column definition used is set up with a reference to the column definition with which the comments were planned (see Allevo settings on SAP side).

Comment fields can also be used to store information that is not directly meant as a planning comment (for example, to control Excel formulas). In this case in particular, you must ensure that the appropriate Excel cell format is used (e.g. "Text" instead of "Standard"), so that, for example, the content is processed reproducibly in Excel (equal signs as the first letter would, depending on the case, indicate the content as a formula or as text content).

The row comments can be included in the Allevo report/report interface; they are therefore also available for evaluations via standard SAP reports (transaction GR55) by means of a document jump. See the Allevo & SAP manual for setting up the interface.

### Remove comments

Usually, only Excel cells with contents are transferred to SAP. This function, which is useful in most cases, can lead to a situation where existing comments cannot be deleted later in SAP (when reading the data again, the comments remain). This situation can occur if the corresponding planning column is empty or the comments are generated via a column definition that is only designed for comments.



Solution for this special case: if the key 'WriteEmptyComments' is active in the Allevo settings on the Customizing sheet, Allevo writes the special character '~' into a comment cell as soon as the user deletes the content (only applies to columns with the suffix '\_N'). Since this special character is also transferred to SAP, it indicates that a comment already existed.

### Sheet comments (per object)

In addition to row comments, Allevo also makes it possible to create a long text comment for each Excel sheet (sheet comment). When calling via an individual object or via MultiPage, the text for the selected object (e.g. cost center, WBS element) is created and saved. A cell with range name CC\_COMMENT must exist on the relevant Excel sheet as a cell definition (see Section 6.2).

### Superordinate display / editing

A superordinate evaluation and editing of comments is also possible via Shuttle or the Allevo Menu |Satellite tables| (see Allevo & SAP manual).

#### 7.5.6 Yearly planning using the “Balloon” variant

As of Allevo 3.4, a three-row COL\_Y range can be used. In this case, the value keys mentioned above are used as suffixes for a column definition (even in activity type dependent planning).

Example with excerpt from an Allevo Master:

|            |       |        |        |       |         |         |         |         |          |          |           |  |
|------------|-------|--------|--------|-------|---------|---------|---------|---------|----------|----------|-----------|--|
| 01         |       |        |        |       |         |         |         |         |          |          |           |  |
| YEAR_WRITE | MO_AO | CY_KEY | CY_ADP | CY_W1 |         |         | CX_WW_Q | CX_WW_V | CYA_0_QF | CYA_0_VF | CYA_0_DKF |  |
| YEAR_READ  | MO_AO | CY_KEY | CY_ADP | CY_R1 | CY_RW_Q | CY_RW_V |         |         |          |          |           |  |
| YEAR_3     |       |        |        |       |         |         |         |         |          |          |           |  |

COL\_Y

Figure 7.6: Example with the Balloon variant for years

Some properties:

- In this case, it is possible to work completely without a suffix: Allevo then independently recognizes on the basis of the respective row definition whether quantities or values are to be read (duplicate columns may no longer be necessary); see column CY\_W1/CY\_R1 in the figure above.
- The third row serves as a trigger for the changed functions (equivalent to the period display); entries in this row are not currently supported.
- The entries for activity type-dependent column definitions are formatted as follows: CYA\_[0-20]\_[COLTYPE], for example, "CYA\_1\_V" (see column marked [1] in the figure above).

### 7.6 Column definitions/Timesets on a monthly basis: Use cases

#### 7.6.1 Differences between the “Compact” and “Balloon” variants

The controlling element of monthly planning functions according to the same principle as yearly planning. The Excel range definition COL\_M must cover at least all columns that deal with monthly data.

For planning on a monthly basis, the three-row flexible “Balloon” procedure has been available since Allevo 3.0. For the older more limited approach, please see the Allevo 2.9 manual.

The “Balloon” variant can plan any number of periods in any number of years. A column differentiation between quantity and value columns is possible, but not necessary in every case (the rows can also determine the differentiation between quantities/amounts).

SAP recognizes the special periods 13 to 16, which can be read and planned explicitly and separately from the 12 months in the Balloon variant.

There is no special range for activity type dependent planning, since a monthly breakdown would result in an unmanageable number of columns. However, you can still do activity type dependent planning on a monthly basis. The various activity types are then displayed at the row level and entered in the column with the definition CM\_KEYRA.

**Note:** Monthly planning can be set up separately from annual planning. Only the ranges ROW\_Y (for sheet recognition) and CX\_WW for Allevo status management are always mandatory.

### 7.6.2 Key columns for monthly planning

In the Excel range definition COL\_M, the cells that identify the columns with the corresponding contents need to be provided with a column definition. The table below shows the column definitions for the key columns:

| Column definition for COL_M |          | Column function  |
|-----------------------------|----------|--|
| Read                        | Plan     |  |
| CM_KEY                      | CM_KEY   | SAP key for cost element, activity type, statistical key figure of group (e.g. cost element group)   |
|                             | CM_KEYRA | Receiving activity type of the cost center   |
|                             | CM_KEYS  | Sender cost center   |
|                             | CM_KEYSA | Sender activity type   |
|                             | CM_FV1   | Variable indicator, if activity type dependent planning is to be used for month zone 1 (V for variable, F for fix)   |
|                             | CM_FV2   | Variable indicator, if activity type dependent planning is to be used for month zone 2 (V for variable, F for fix, opposite to CM_FV1)   |
|                             | CM_ADP   | Indicator whether the cost element or statistical key figure under CY_KEY should be planned as activity type dependent. An entry of 1 activates activity type dependent planning via Excel, so that the settings should be set up in accordance with Section 7.5.2. If you enter 3, planning is carried out using the set of rules from rule planning (see Allevo & SAP manual). |

**Note:** The key columns of annual planning and monthly planning, which are identical in terms of content, can also be arranged on top of each other, as shown in Figure 7.2 for the cost element.

### 7.6.3 Setting up monthly planning

The functional principle of the Allevo column definitions for monthly planning differs slightly from that of annual planning. Here, although each column on the Excel side - that is, each individual month - has its own definition, the Allevo administration menu on the SAP side contains the definitions for the year and the version for the twelve months as a whole. The "Balloon" variant for the

months is defined by a three-row range "COL\_M", whereby the third row directly corresponds to the respective period.

|       |       |       |       |       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 | CM_W2 |
| CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 | CM_R2 |
| 1     | 2     | 3     | 4     | 5     | 6     | 7     | 8     | 9     | 10    | 11    | 12    |

Figure 7.7: "Balloon" variant with three-row range COL\_M

|              |  |
|--------------|--|
| <b>Note:</b> | Monthly planning only contains a single column per month by default. Using the key extensions _Q/_V, a further column can be added for differentiating between quantity and value. |
|--------------|--|

Within the first month block (CM\_R1), actual and plan data can be mixed during the readout (for example, months 1-7 actual data, as of month 8 plan data; see also F1 help for the constant YEAR1\_PERIOD):

- 1<sup>st</sup> month block, i.e. for months 1 to 12 and the column definition CM\_R1:
  - Plan: CX\_WW
  - Read: CX\_RR, CM\_R1 (depending on activation of the mixed function)

The same row definitions as for yearly planning are used (see Section 7.7.2).

You can use the extensions known from annual planning to define columns (suffix \_V, \_Q etc., see Section 7.5). The column definition can also be entered without a suffix, as shown in the figure above. Allevo then decides on the basis of internal characteristics whether the entry is to be treated as a quantity or a value (i.e. quantity for statistical key figure and activity type, value for primary costs, etc.).

|              |  |
|--------------|--|
| <b>Note:</b> | Please note that the Allevo column definitions CX_RR and CX_WW, which are used to define the settings for reading actual data (current year) or for planning in the main planning year, are also used for monthly planning. Be sure to note constant YEAR1_PERIOD.   |
|              | It is also possible to place the Allevo column definitions for reading and planning by year or by month next to each other - that is, in separate columns - instead of using the definitions in the same column. For example, you can use formulas to calculate new plan values from read values.  |
|              | Pro tip: If the values or quantities for monthly planning are determined by formulas, you must use the formula ROUNDING to two decimal places to ensure that the quantities/values can be written to SAP. Quantities or values in yearly columns, on the other hand, are rounded automatically. Make sure that the 12th month column is used as a correction column (annual value minus periods 1-11) to check for any rounding errors that may have occurred. |

#### 7.6.4 Balloon variant with monthly blocks one below the other

By default in Allevo, the monthly values are displayed in blocks next to each other. When there are at least 3 monthly blocks next to each other, it becomes difficult to see the whole picture on the screen. In this case, an alternative display of monthly planning with one row per year (one below the other) can be useful.

The following figure shows a schematic of the structure, including the necessary ranges.



|      |    |   |   |  |       |  |      |  |
|------|----|---|---|--|-------|--|------|--|
| 1000 | KS | K | X |  | CM_R3 | <b>Activity Output</b>                 | CC_X |  |
|      |    |   |   |  |       |  |      |  |
| 1000 | KS | C | Z |  | CM_R3 | <b>Activity Input (Partner Object)</b> | CC_Z |  |
|      |    |   |   |  |       |  |      |  |
| 1000 | KS | C | Z |  | CM_R3 | <b>Activity Input (Source Object)</b>  | CC_Y |  |

Figure 7.9: Balloon variant with single rows per year in the dynamic range

The following setup steps are necessary for the Allevo Master (see also the corresponding marked areas in Figure 7.8 and Figure 7.9).

- Enter column ID (2.1 in Figure 7.8).  
In this case DYN\_COLDEF is used as the column ID; this should be entered in the range COL\_Y (first row for planning, second row for reading).
- Enter row key (2.2 in Figure 7.9):  
Below the entries for DYN\_COLDEF, the appropriate column definitions should be entered again. Important: the creation of dynamic rows is linked to annual values; therefore, row keys must be entered in the Row\_Y range.

Since the dynamic rows are compiled using functions on the SAP side of Allevo, the constant DYN\_COLDEFS must be set: it determines the row definitions for which data is to be read and which column definitions are relevant (see the documentation for the constant).

## 7.7 Row definitions / AllocationSets

### 7.7.1 Operating principle

After the column functions (or more generally the TimeSets) are defined by entering the Allevo column definitions, the behavior of individual rows is further controlled via the Excel ranges ROW\_Y and ROW\_M. ROW\_Y and ROW\_M mark the Excel cells in which the necessary Allevo row definitions are entered (for annual or monthly planning). The ranges ROW\_Y and ROW\_M always consist of three contiguous columns (Write/Read/Read2).

Main area of the cost element structure

| 1000                          | Kantine                          | EC | ADP  | Sender | Empfänger | Actual 1-12 2012 |            | FC 1-12 2014 |               |          |       |          |       |
|-------------------------------|----------------------------------|----|------|--------|-----------|------------------|------------|--------------|---------------|----------|-------|----------|-------|
| Hertwig                       |                                  |    | Year | Month  | Object    | Activity Type    | Objecttype | Object       | Activity Type | Quantity | Value | Quantity | Value |
| <b>Statistical Key Figure</b> |                                  |    |      |        |           |                  |            |              |               |          |       |          |       |
|                               | 2010 Square meter                |    | A    |        |           |                  |            |              |               |          |       |          |       |
|                               | 4510 Power usage, drying process |    | S    |        |           |                  |            |              |               |          |       |          |       |
| <b>Activity Type</b>          |                                  |    |      |        |           |                  |            |              |               |          |       |          |       |
|                               | 1410 Service Hours               |    |      |        |           | 1                |            |              |               |          |       |          |       |
| <b>Cost Element</b>           |                                  |    |      |        |           |                  |            |              |               |          |       |          |       |
|                               | 420000 Direct labor costs        |    | 1    |        |           |                  |            |              |               |          |       |          |       |
|                               | 420001 Direct labor costs        |    | 1    |        |           |                  |            |              |               |          |       |          |       |
|                               | 421000 Indirect labor costs      |    | 1    |        |           |                  |            |              |               |          |       |          |       |
| #                             | Wages                            |    |      |        |           |                  |            |              |               | 0        | 0     | 0        | 0     |

Figure 7.10: Excel range ROW\_Y

The Allevo row definitions determine what should happen in the individual rows where cost elements, statistical key figures, and activity types could be found: The Allevo row definition determines the planning and read mode of these rows.

- A row definition should be entered in the first column (write) of the Excel ranges ROW\_Y and ROW\_M, respectively, if the statistical key figure, activity type or cost element should be planned on a yearly or monthly basis.
- In the second column (read), a row definition should be entered if a statistical key figure, activity type or cost element should be read.
- In the third column (read2), an entry is made for the respective Allevo row definition if an additional read operation is required for the cost element. This can occur if a primary cost element was posted using a primary or secondary transaction. In this case, for example, you must enter the Allevo row definition **A** in the second column for reading the primary posted costs and the Allevo row definition **I** (internal) in the third column for reading the secondary posted costs.

Important: all values from "read" and "read2" are added together. In current Allevo versions, the contents of the line item list are also determined according to the same criteria: selection in two steps with the attributes of the individual row definitions (for example, when taking into account the relevant transactions). If these restrictions do not apply for the line item, you can set the constant NO\_LINEITEM\_BUSTRANS.

Comments are not read with Read2.

The Excel ranges ROW\_Y and ROW\_M go from the first to the last row that contains a key for a cost element (or statistical key figure or activity type).

**Note:** This range can also contain rows that are irrelevant for direct communication with SAP, e.g. help or summary rows. These rows are then not provided with an entry for an Allevo row definition

### 7.7.2 List of Allevo row definitions

The following row definitions apply for both the yearly and the monthly ranges. They can be used as a definition for planning and reading out data (read and read2): However, individual row definitions are restricted to reading or writing and cannot be used for all object types (see list below).

**Note:** Since there is only one definition for the entire row in monthly planning, all months can be seen as a single unit (see also constant YEAR1\_PERIOD and YEAR2\_PERIOD). It is therefore sufficient if a single period is specified in Excel for posting all the months (the rest would be transferred to SAP with 0). However, the safest option is to transfer between Excel and SAP every 12 months.

The following table shows the ID-code for the row definition in the Allevo Master and the corresponding planning area. The list here primarily refers to use in CO overhead cost planning. For use within the framework of profit center accounting, see the Allevo Profit Center manual.

| Planning area                  |   | Necessary Key  | Function |      | Usable for: |    |    |    |    |
|--------------------------------|---|--|----------|------|-------------|----|----|----|----|
|                                |   |  | Plan     | Read | KS          | OR | PC | PR | BP |
| <b>Statistical Key Figures</b> |   |  |          |      |             |    |    |    |    |
| R                              | Summary values  | Statistical key figure   | yes      |      | X           | X  | X  | X  | X  |
| Q                              | Fixed values  | Statistical key figure   | yes      |      | X           | X  | X  | X  | X  |
| S                              | Fixed values (average)  | Statistical key figure   | yes      |      | X           | X  | X  | X  | X  |
| <b>Activities</b>              |   |  |          |      |             |    |    |    |    |
| M                              | Activity quantity (see also constant ACT_SCHED_MV)  | Activity type  | yes      | yes  | X           |    |    |    |    |
| N                              | Capacity  | Activity type  | yes      | yes  | X           |    |    |    |    |
| <b>Costs</b>                   |   |  |          |      |             |    |    |    |    |
| A                              | Primary costs, if necessary also revenues if the value type for statistical postings is active. | Cost element   | yes      | yes  | X           | X  | X  | X  | X  |
| B                              | Same as A (primary costs), quantities are also taken into account (plan and read)               | Cost element   | yes      | yes  | X           | X  | X  | X  | X  |
| C                              | Activity allocation   | Cost element,<br>Sender-KOSTL,<br>Sender-LA  | yes      | yes  | X           | X  | X  | X  | X  |
| D                              | Activity allocation with prices   | Cost element,<br>Sender-KOSTL,<br>Sender-LA  | yes*     | yes  | X           | X  | X  | X  | X  |
| E                              | Activity allocation, credit   | Cost element,<br>Sender-LA   | no       | yes  | X           |    |    |    |    |
| K                              | Activity allocation, credit   | Cost element,<br>Sender-LA,<br>Partner-cost center   | no       | yes  | X           |    |    |    |    |
| F                              | Settlement primary  | Cost element   | yes      | yes  | X           | X  | X  | X  | X  |
| G                              | Settlement secondary  | Cost element,<br>Sender-LA,<br>Sender-KOSTL,<br>(as of 3.5 empty cost centers are accepted as a sender). | yes      | yes  | X           | X  | X  | X  | X  |

## Main area of the cost element structure

| Planning area                                   |  | Necessary Key  | Function |      | Usable for: |    |    |    |    |
|---|--|--|----------|------|-------------|----|----|----|----|
|   |  |  | Plan     | Read | KS          | OR | PC | PR | BP |
| H   | Settlement secondary, with prices  | Cost element, Sender-KOSTL, Sender-LA                                  | yes*     | yes  | X           | X  | X  | X  | X  |
| I   | Secondary costs, general   | Cost element   | no       | yes  | X           | X  | X  | X  | X  |
| J   | Secondary costs, same as I, but quantities are read as well  | Cost elements (limit to partner objects with constant PARTNER_FROM_TO) | no       | yes  | X           | X  | X  | X  | X  |
| L   | Delimitations from COSB  | Cost element   | no       | yes  | X           | X  | X  | X  | X  |
| O   | Secondary costs, special   | Cost elements, Partner object via constant                             | no       | yes  | X           | X  | X  | X  | X  |
| <b>Profit-Center-Integration (all accounts)</b> |  |  |          |      |             |    |    |    |    |
| P   | Account for profit center (reading via all company codes; planning via automatic entry to reconciliation account incl. quantities) | Cost element (account)   | yes**    | yes  | X           | X  |    | X  | X  |
| U   | Account for profit center (direct posting to PC)***  | Cost element (account)   | yes      | no   | X           | X  | X  | X  | X  |
| <b>Dynamic ranges</b>                           |  |  |          |      |             |    |    |    |    |
| X   | Dynamic crediting (s. Section 7.7.3)   | none   | no       | yes  | X           | X  | X  | X  | X  |
| Y   | Dynamic settlement   | none   | yes      | yes  | X           | X  | X  | X  | X  |
| Z   | Dynamic activity allocation  | Cost element, partner object to/from (according to constant)           | yes      | yes  | X           |    |    |    |    |
| <b>Comments</b>                                 |  |  |          |      |             |    |    |    |    |
| T   | Comments   | Cost element or free entry   | yes      | yes  | X           | X  | X  | X  | X  |

\* No planning of prices

\*\* automatic entry to reconciliation account of the profit center via the standard plan integration of cost centers, orders, or WBS elements in SAP, i.e. an order or WBS element is posted in the background (automatic entry rules in Allevo must be maintained).

\*\*\* Direct posting of the profit center on the account level. The profit center and company code that are stored for the starting object apply here. In this case, data are also read via row definition P.

|              |   |
|--------------|---|
| <b>Note:</b> | All row definitions can be copied and individualized in terms of their reading capabilities via the Allevo menu  Settings , folder  Row definitions . In this way, in addition to definition <b>A</b> , it is then possible to create a definition <b>A1</b> , which differs from <b>A</b> with regard to the combination of business transactions, value types and credit and debit indicators |
|              | Please see the Allevo & SAP manual and the F1 help for this menu for more details.  |
|              | For activity type dependent planning, the same row definitions can be used. In order to activate activity type dependent planning, use the column CY_ADP (also CM_ADP) for which the  |



|  |  |
|--|--|
|  | <p>type of activity type dependent planning should be set for each cost element or statistical key figure:</p> <p><b>1</b> = activity type <b>dependent</b> planning via Excel, including reading (the activity type indicated on the Excel side!)</p> <p><b>2</b> = activity type <b>dependent</b> planning via Excel, only planning (reading proceeds in an activity type <b>independent</b> manner!)</p> <p><b>3</b> = activity type <b>dependent</b> planning using rules, including reading (the activity type indicated on the Excel side!)</p> <p><b>4</b> = activity type <b>dependent</b> planning using rules, only planning (reading proceeds in an activity type <b>independent</b> manner!)</p> |
|  | <p>If desired, Allevo can read quantities independently of the partner object stored in the Master. The summation proceeds independently from the partner object (see constant SELECT_NO_PAROB).</p>   |

### 7.7.3 Dynamic rows (dynamic range)

In general, the information in the Allevo Master determines which cost elements, activity types or statistical key figures are processed in planning.

However, there are some situations in which it is not simple to provide the necessary information. In this case, we are dealing with the planning of secondary costs from

- Direct activity allocation (debit) and
- Secondary order settlement

In such cases, it is necessary to provide information on the sender cost center and sender activity type during planning. The number of possible combinations can be so large and variable depending on the receiver, that it is virtually impossible to create a valid Excel form in advance.

A similar circumstance arises with

- credits from activity allocation

Here, the receiving partner is relevant. Since we are dealing with a credit from activity allocation that is to be carried out in the plan via activity planning, this range is only available for reading, but not for planning (for other use cases such as "FI Dynamic", see below).

In such cases, Allevo offers the dynamic range function. After selecting a cost center (or order or WBS element), only the sender cost centers/sender activity types relevant for the selected object are determined from past postings and inserted into the Allevo Master. This is done by automatically inserting rows in the Excel planning layout with the disclosure of the cost element, sender cost center, and activity type.

The dynamic ranges in the Allevo Master consist of two rows each. On the one hand, the rows should be identified by an Excel name, which must be entered in a cell of the column of the object key (CY\_KEY). On the other hand, the rows require the corresponding Allevo row definitions for reading (see Figure 7.11). The following assignments apply:

| Excel Name (Object-Key) | Recommended row definition | Comment   |
|-------------------------|----------------------------|---|
| CC_X                    | K                          | Activity Output (credit from activity allocation to cost centers)<br>Row definition K makes it possible to read values for activity allocation, as of version 3.4, planning is also possible (see note below).          |
| CC_Z                    | C                          | Activity input (Partner object / from direct allocation)  |
| CC_Y                    | G                          | Activity input (source object / via settlement)<br>As the result of a summary settlement via multiple objects there can also be entries with an empty source object; displayed as a separate row in the Allevo-Dynamic. |

The following figure shows the corresponding section of the Allevo Master.

Figure 7.11: Dynamic ranges in the Allevo Master (outlined in green).

The cells outlined in red in the figure above are automatically filled. In addition, the cost elements are placed in the cells in the column AG (CY\_KEY/CM\_KEY), the previously entered pseudo-object key is replaced by the cost element.

For dynamic determination of the relevant rows, the corresponding row definitions X, Y, and Z are used (usually entered in the column "Read2").

When reading reference data for the first time, Allevo determines the posting combinations that already exist in the system. The system inserts the calculated rows with details on the cost element, sender cost center, and sender activity type at the place where only one row existed before. The Allevo line definitions X, Y or Z are deleted: after this first run the dynamic is "switched off".

|              |   |
|--------------|---|
| <b>Note:</b> | The dynamic ranges should be located at the bottom end of the planning range for yearly planning. Complete rows should be inserted, extending the range of activity type dependent planning and monthly planning. |
|              | The cell definitions CC_X, CC_Y and CC_Z are to be used as Excel names and as normal entries in the cells. If the codes are not in the cells named above, Allevo does not determine any posting combinations.     |

When data for the dynamic range is read again, not all fields are updated (for example, when reading/writing comments using row definition T). The requirements may have to be clarified in the implementation project.

### Use areas

The dynamic rows are created with reference to the Allevo start object, for example, the cost center. If a 1:n group is stored there, the dynamic rows are determined as aggregated for all objects of the corresponding group: that is, separated by sender and receiver activity types, but not separated by object (for this case in particular, see the notes on performance below).

In MultiPage mode, the reference to the object applies per sheet. Flexible starting via group or object range in MultiObject mode has been supported since Allevo 3.4. In earlier versions, use together with MOD was only possible when starting with a single object.

### Special features for activity output (range X)

The dynamic range X makes it possible to view the posting relationships from the sender's point of view (activity output). Reading takes place via row definition K, for versions 3.5 and higher, planning is also possible via these rows (in earlier versions only via FP with allocation type DA).

#### Note:

The term "activity output" is used in Allevo, but in fact with row definition K a quantity taken is posted as with KP06 (view 1-102).

For planning, please note:

- The relevant planning information must be entered in the Allevo Master (especially details on the receiver, see the next note).
- Allevo automatically changes the sign (+/-) of the activity quantity.
- The Allevo status for the receiver object is handled according to the plan year/version in the current layout (usually, for example, the receiver cost center must be active in status management).

### Receiver display

In standard Allevo, the receiver object is written into a single cell (e.g. column CY\_KEYR). This is also the case for KL object types.

In some use cases, it may be useful to separate the receiver information by cost center and activity type. For example, when planning via row definition K or executing posting via FP. For a separated display, the constant SPLIT\_PAROB must be active: Allevo then uses the columns CY\_KEYR, CY\_KEYRA and KEYRTYPE (Note: for CY\_ADP > 1, CY\_KEYRA cannot be read/planned).

The separated display is currently only possible for dynamic range X and row definition K. As a second function, the constant SPLIT\_PAROB now also allows the entry of special objects that are not normally supported by Allevo (see F1 documentation).

Allevo also supports the display of cumulative values in the X range. Two approaches are available:

- When using the constant DYN\_X\_EXT, the rows of several business transactions per sender and receiver object are summarized and then, for example, planned collectively via an activity type.

Different criteria can be defined: for example, cumulated at cost element level or sender/receiver activity type (see the documentation for the constant).

- If the SAP document update is active, the totals table COSS also contains many entries (for example, one data record per order). With OR as the partner object type, the list in the dynamic range of Allevo (CC\_X) is then also so long that it can hardly be used as the basis for further planning. In this case, DYN\_X\_AGGR can combine the rows.
- For activity input with combination of different senders, prices are averaged.

### Display of receiver for activity input (CC\_Y / CC\_Z)

By default, the receiver is completely written in the field CY\_KEYR. However, it can be advantageous for downstream functions if the corresponding specifications are entered separately in Excel according to object type, object and activity type (column key CY\_KEYRTYPE, CY\_KEYR and CY\_KEYRA). To do this, constant USE\_DYN\_ACTDEP must be set.

### Performance / additional filters

Using the dynamic range can impede the performance, since searching is then done without a specification of cost elements or sender objects. The Allevo menu |Settings|, folder |Dynamic ranges| lists the cost elements to be searched for, under which the activities of activity allocation and order settlement take place.

Optionally, a restriction can be made via the receiver activity type. This may be useful if, for example, only some of the activity types are relevant for planning.

### Special cases when using dynamic ranges

- FI dynamic (FID)

This Allevo function enables you to dynamically read 'external activity relationships' with vendors and customers in the same way as the dynamic of 'internal activity relationships'. The settings required to make the data available mainly affect the SAP side and are therefore described in the Allevo & SAP manual.

- Transfer partner name from SAP  
The function of comment columns in Allevo allows you to read descriptive texts of involved objects and pass them to the Allevo Master. This function is controlled by the constant READ\_PARTNER\_TEXTS. It can be particularly helpful in the dynamic range to display a descriptive text for the sender and/or receiver.
- Monthly planning with individual rows per year  
Monthly planning allows you to work with one row per year (instead of column blocks). This application requires a different reading procedure and is described above (see Section 7.6.4).

### Constants for dynamic rows

The operation of the dynamic ranges can be controlled with the following constants:

- DYN\_X, DYN\_Y, DYN\_Z Deactivate the SAP reading function for the respective range
- DYN\_X\_AGGR Dynamic X with aggregation to partner object OR
- DYN\_X\_EXT Dynamic X cumulation via activity type, cost element, or cost center
- DYN\_X\_PC Replace CO-receiver with profit center, with aggregation if necessary
- SPLIT\_PAROB Split KL-Object for activity type dependent planning

- USE\_DYN\_ACTDEP

See the F1 documentation for further details on individual constants.

#### **7.7.4 Additional functions for reading reference data**

The row definitions mentioned above determine the standard behavior of Allevo, especially for reading reference data. Individual adjustments are possible via corresponding settings on the SAP side (e.g. for value type, business transaction), or via additional constants.

Examples:

- The constant READ\_WITH\_KEYR makes it possible to read data restricted to functional area.
- The reading of prices for secondary costs is controlled via the constant READ\_SEC\_TARIF.
- Constant READ\_DISKEY determines how the distribution key is to be handled from year to period.

## 8 Satellite ranges / SatelliteCorner

Allevo makes it possible to transfer data from up to 99 independently definable Excel tables to SAP and store it centrally in corresponding SAP tables. For detailed information, see the Allevo & SAP manual.

On the Excel side, the data is always displayed or maintained depending on the selected object. In SAP, the data is stored in the same table for all object types, objects, versions, and fiscal years.

|              |  |
|--------------|--|
| <b>Note:</b> | Satellites must be active in the respective constants for the planning layout so that Allevo takes them into account when reading and writing (constant ACTIVE_SAT). |
|--------------|--|

The data in an SAP satellite table are displayed in Excel via two specific ranges, the row and column axis of the respective "SatelliteCorner".

### 8.1 Control elements of the SatelliteCorner

The SatelliteCorners in the Allevo Master serve to allow the transfer of data into/from an SAP satellite table. For each corner, two axes must be defined as Excel name ranges:

- A **column axis** controls, e.g. field mapping, or the assignment of Excel columns to the fields of the SAP table and optional functions using these KeyPointers:
  - READ and WRITE: Assignment of the columns to be transferred from/to SAP using either the key names of the individual satellite fields or their field position in the SAP table.
  - FORMULA: Ability to store Excel formulas, which are copied into the corresponding columns after reading the data
  - FORMULATOVALUE: Removing formulas in certain columns to maintain their calculation results as a value (performance)
  - SORTING: Specification of further key criteria next to the ROW field

Field number 7 / ROW of the satellite table has a special meaning. The satellite ranges on the Excel side usually begin with this field; when working with more than one object, further key fields of the satellite table (field numbers 1-6) should also be taken into account. The following fields (starting with Nr. 7) are implemented via an append for the satellite table (see Figure 8.1).

- The control element for the **row axis** marks the data range of the satellite and thus the cells in which data is exchanged with the satellite table.
  - ROW: marks the row under which the data range begins. As a rule, the range ends with the bottom end of the row axis (exception in insertNewSatRows mode, in which the data range is searched even for additional rows).
  - KEY: marks the assignment of the row number (field Nr. 7); The number of rows in this range is usually interpreted according to the expected data volume. If this estimation is difficult, Allevo provides a dynamic variant in which rows can be copied automatically (possible for entire row or only satellite range).

On the SAP side, there are 99 satellite tables available. The following Excel ranges are used for controlling data transfer:

| Nr. | Name of the satellite table on SAP side | Header name for field mapping | Name of data range |
|-----|---|-------------------------------|--------------------|
| 1   | /KERN/IPPSAT01                          | SAT01HR                       | SAT01RW            |
| 2   | /KERN/IPPSAT02                          | SAT02HR                       | SAT02RW            |
| ... | ...                                     | ...                           | ...                |
| 99  | /KERN/IPPSAT99                          | SAT99HR                       | SAT99RW            |

The Excel ranges named so far relate to the standard use cases; for special write or read functions, additional range names are available (see Section 8.5).

The following figure shows the relationship between the satellite tables in the SAP system and the representation via the Excel ranges.

The axis ranges formatted in blue are defined as name ranges in Excel (vertical: SATxxRow, horizontal: SATxxColumn). The functional assignment of the respective columns and rows in these ranges (through KeyPointer) are highlighted here in green and yellow. The uppermost arrow shows the connection between the satellite number in Excel and the table key in SAP. The other arrows show the referencing of the table fields in Excel (here as key name). The numbers outlined in red indicate the number of the field position, which can be referenced alternatively in Excel.

The figure shows a side-by-side comparison of an Excel spreadsheet and the SAP Dictionary. On the left, an Excel spreadsheet titled 'Satellite01' has columns 'ROW', 'KEY', 'PROJECT', and 'INV\_CAT'. The 'KEY' column is highlighted in green, and the 'PROJECT' and 'INV\_CAT' columns are highlighted in yellow. On the right, the SAP Dictionary 'Dictionary: Tabelle anzeigen' shows the structure of the table '/KERN/IPPSAT01'. Fields listed include MANDT, KOKRS, .INCLUDE, .APPEND, .INCLUDE, PROJECT, and INV\_CAT. Red arrows point from the Excel 'KEY' column to the 'MANDT' field in the dictionary, and from the 'PROJECT' and 'INV\_CAT' columns to their respective fields in the dictionary. Red circles with numbers 7, 8, and 9 are placed next to the 'ZEILE', 'PROJECT', and 'INV\_CAT' fields in the dictionary, indicating their field positions.

Figure 8.1: Satellite tables, mapping on the Excel and SAP sides

|              |  |
|--------------|--|
| <b>Tip:</b>  | As long as the field name is directly entered in the header, the assignment is not lost when changing the structure of the append (field sequence!)  |
| <b>Note:</b> | The procedure to be followed (name or number) must be valid for the header; a mix of both specifications is not possible. Allevo decides on the basis of the entry in column no. 7 (i.e. the field name ROW can be in column no. 7). |

The first column in the figure above has a special meaning as a row number: it is usually numeric and is related to the currently edited object (such as cost center) and the other criteria such as plan year and version. If the Allevo Master only works with one object (or a representative object), then the row number is a unique characteristic for the data record of a satellite table (it may also be stored with this number in the corresponding satellite table in the SAP system).

For this reason, in many cases, column no. 7 of the satellite table is also created as the first column in the respective area. When reading data from SAP, the number is usually copied from the corresponding column 7 (= ROW field) of the satellite table. Using special functions, however, it is also possible to insert the data into Excel with row accuracy (see notes below).

Excel formulas are not overwritten when reading data from SAP. Instead, you can restrict the use of formulas to actually filled rows (this can improve performance, see Section 8.7).

## 8.2 Read and write satellite ranges (sequential)

In most applications, data from a satellite table is sequentially inserted (or collected) in the corresponding satellite range. The order of insertion is defined by column 7 (= row). In Excel, data is only entered in rows for which a row number has already been entered. This standard case also corresponds to the illustration in Figure 8.1 above.

For more complex applications of Allevo with different objects or object types on a single sheet, the data of the satellites for these different characteristics must also be read or written (e.g. controlled by the Allevo constant GRP\_READ\_SATxx). In this case, the row number is no longer unique and additional columns from the satellite table index must be displayed in the Excel satellite range. SETCLASS and COBJECT are mandatory.

|              |   |
|--------------|---|
| <b>Note:</b> | <p>In general, all fields of a satellite can be displayed in Excel. Data such as object, year and version are still constant in standard Allevo applications for the entire planning process and are filled automatically when you save in SAP (especially in single and MultiPage-mode with one object per sheet).</p> <p>In MultiObject-mode, index fields 1 to 6 should always be managed on the Excel side, especially when editing cross-object data (see F1 documentation for constants SATxxSELECT or GRP_READ_SAT).</p> |
|--------------|---|

### Saving satellite data

When saving to SAP, only the rows with filled row numbers are transferred to SAP and stored in the relevant satellite table. When saving data in SAP, the column directly after the row number (that is, after column 7) has a special meaning: a row is only saved if an entry exists in the corresponding field of this column. This rule applies regardless of whether this column is selected for communication with SAP via the header area or not.

You can make use of this property if you want to control the saving of a data record (row) in SAP on the basis of other criteria. For example, if these criteria exist, you can use an IF formula in the second column to enter X, for example. If the criteria are not available, an empty entry is set and the row is not saved in SAP.

|                 |  |
|-----------------|--|
| <b>Example:</b> | <p>The second column in Figure 8.2 is the personnel number. If an entry is missing there, the respective row is not transferred to SAP despite there being a row number. In the example above, rows 6, 8 and 9 are not saved in the corresponding satellite table on the SAP side.</p> |
|-----------------|--|

On the SAP side, all satellite data for the current object (as well as the year, version, etc.) is deleted: at the end, the satellite table contains an exact copy of the data that was previously displayed in Excel. In MultiObject-mode, satellite data for different objects / object types are also stored if required (the corresponding fields must therefore also be filled on the Excel side).



### 8.3 Individually design satellite ranges

Satellite data do not have to be strictly sequentially listed, but can be split into different blocks within the read/write ranges (as long as the column layout is continuous).

**Example:** In the satellite range, there may be headings or other rows that are not transferred to SAP at a later time.

The term MatchExistingRow is common for this procedure; the figure below shows the basic structure.

| ... | BA                          | BB          | BC        | BD         | BE         | BF        | BG |
|-----|-----------------------------|-------------|-----------|------------|------------|-----------|----|
| ... |                             |             |           |            |            |           |    |
| 451 | <b>Kopfbereich</b>          |             |           |            |            |           |    |
| 452 |                             |             |           |            |            |           |    |
| 453 | 7                           | 8           | 9         | 10         | 11         | 12        |    |
| 454 |                             |             |           |            |            |           |    |
| 455 | Zeilen-Nr.                  | Personal-Nr | Name      | Eintritt   | Austritt   | Kapazität |    |
| 456 |                             |             |           |            |            |           |    |
| 457 | <b>Angestellte</b>          |             |           |            |            |           |    |
| 458 | 1                           | 4711        | Müller    | 01.01.1976 | 21.12.2008 | 100%      |    |
| 459 | 2                           | 5345        | Maier     | 01.04.1987 |            | 100%      |    |
| 460 | 3                           | 2764        | Schmitz   | 01.06.2001 |            | 75%       |    |
| 461 |                             |             |           |            |            |           |    |
| 462 | <b>Auszubildende</b>        |             |           |            |            |           |    |
| 463 | 4                           | 6538        | Schreiber | 01.08.2007 |            | 50%       |    |
| 464 | 5                           | 3566        | Liebke    | 01.08.2007 |            | 50%       |    |
| 465 | 6                           |             |           |            |            |           |    |
| 466 |                             |             |           |            |            |           |    |
| 467 | <b>Praktikanten</b>         |             |           |            |            |           |    |
| 468 | 7                           | 3456        | Hamann    | 01.08.2007 | 30.09.3007 | 100%      |    |
| 469 | 8                           |             |           |            |            |           |    |
| 470 | 9                           |             |           |            |            |           |    |
| 471 |                             |             |           |            |            |           |    |
| 472 | <b>Schreib-/Lesebereich</b> |             |           |            |            |           |    |
| ... |                             |             |           |            |            |           |    |
|     | ◀◀▶▶                        |             | ALLEVO    |            |            |           |    |

Figure 8.2: Read/write ranges with split contents

Data is only imported into the Excel rows for which a row number is entered in column no. 7. Intermediate rows without a number are not taken into account when reading and writing data from SAP.

**Note:** In the standard case, the data are entered in the sequence in which it comes from SAP. The information in column no. 7 therefore has a more symbolic character.

When writing satellite data, the row numbers are also recounted in the SAP database table before they are stored.

However, the row numbers can explicitly specify which data row is to be inserted from SAP (the MatchExistingRow reading and writing is described in the next sections).

### 8.4 MatchExistingRow: mode for row accurate reading and writing

In the use case described above (see last section), data from a satellite table is also inserted sequentially into the corresponding satellite area. In some applications, however, it is necessary to insert the read data exactly into predefined rows on the Excel side.

The link is made via the specified Excel line number in column 7 and the corresponding number in the SAP satellite table: that is, with the number stored in the ROW field of the satellite table.

This exact assignment allows you to arrange satellite rows in predefined order on the Excel sheet (e.g. 1, 5, 2, 23, ...).

|                 |  |
|-----------------|--|
| <b>Example:</b> | The row number can be determined from other parameters on the Excel side: for example, using a formula with reference to cost element to create a satellite data row exactly above the Excel row with the appropriate cost element (in current Allevo versions max. 10 digits are possible). |
|-----------------|--|

The function for this exact form of reading and writing is activated by entering MatchExistingRow in the uppermost left cell of the intersection of the two satellite axes.

### Exact row reading for multiple objects simultaneously

When working with different objects or object types on a single sheet (MO-mode) the data of the satellites for these different characteristics must also be read or written.

In this case, column 7 is no longer sufficient for exact positioning; instead, SETCLASS and COBJECT specifications must be included in the positioning. For this purpose, this additional information can be configured via the KeyPointer SORTING. Here's the basic structure:

|                    |     |          |         |       |     |
|--------------------|-----|----------|---------|-------|-----|
| <b>Satellite01</b> |     |          |         |       |     |
| MatchExistingRow   | ROW |          |         |       | KEY |
| WRITE              |     | SETCLASS | COBJECT | ZEILE |     |
| READ               |     | SETCLASS | COBJECT | ZEILE |     |
| SORTING            |     | SETCLASS | COBJECT |       |     |
| FORMULA            |     |          |         |       |     |
|                    |     |          |         |       |     |
|                    | X   |          |         |       |     |
|                    |     |          |         |       |     |
|                    |     |          |         |       |     |
|                    |     |          |         |       |     |
|                    |     |          |         |       |     |
|                    |     |          |         |       |     |

|  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

Figure 8.3: Row exact reading via multiple objects

On this basis, Allevo attempts to find the correct row on the Excel side when reading each row of a satellite. The following determination rules apply (in this sequence):

- First in the SORTING mapped field (e.g. Setclass)
- Second in the SORTING mapped field (e.g. CO object)
- Column with header index 7 (or corresponding field name).

### 8.5 Other execution modes for special read and write functions

The method of data transfer between Excel and SAP can take place in different ways and is defined for each satellite at the upper left-hand intersection.

The following shows an excerpt from the Allevo Master:

| SatelliteXX    |       |  |          |
|----------------|-------|--|----------|
| Standard       | KEY   |  |          |
| WRITE          | ZEILE |  | ZPROJECT |
| READ           | ZEILE |  | ZPROJECT |
| FORMULA        |       |  |          |
| FORMULATOVALUE |       |  |          |

| SatelliteOperations |  |
|---------------------|--|
| Mode                | Remark   |
| Standard            | All satellite rows must exist in the Allevo Master                                     |
| MatchExistingRow    | Like <i>Standard</i> with key matching (Excel is master).                              |
| InsertNewSatRows    | Only 2 rows must exist in Excel. Copying is done locally only per each satellite area. |
| InsertNewExcelRows  | Only 2 rows must exist in Excel. Copying is done for the full excel row.               |

Operation Mode

Figure 8.4: Section from the definition sheet "Satellite" in the Allevo Master

**Standard:** The standard rules mentioned above apply, according to which the Excel range has to be prepared in advance for the expected number of rows.

**MatchExistingRow: row exact reading and writing**

In this procedure, Allevo writes the data of a satellite exactly into the Excel row, which is also stored in the SAP satellite table: that is, with the number in the ROW field of the satellite table (the 7th field of the table). Details on this application are described in Section 8.4.

**insertNewSatRows**

**insertNewExcelRows**

**Additional name range RWS: Shuttle Variant**

This variant was developed for working with the additional module "Shuttle", because this module makes it possible to read and write satellite data over several objects at the same time. For this purpose, it is also necessary to display the contents of all index fields of a satellite table in the Master (i.e. additionally columns 1 to 6, the fields KOKRS to VERSION of the satellite table).

For this application in the Shuttle, a range SATxxRWS is set up in addition to SATxxRW. The range should be exactly above the first row of SATxxRW (the number of rows is still determined by SATxxRW).

**Note:** The SATxxRWS range is now only required for use with Shuttle; working with index columns in the Allevo Master is now also possible via SATxxRW.

For details on using this range, see the Allevo Shuttle manual.

**8.6 Number of rows in the data range**

With Allevo Standard, the number of rows in the data range of Excel is interpreted according to the expected data scope. This can be difficult in individual use cases (for example, when dynamically selecting data using MOD or ProCED).

The following Allevo functions can be helpful in this case:

- Change to the mode insertnewSatRows or insertNewExcelRows: this allows the row range to be dynamically expanded – it is dependent on the number of rows transferred from SAP. Formulas are also copied (see the corresponding parameter in Customizing).

- As an alternative, Allevo also provides a control function which is executed each time data is transferred from satellite data to Excel. If there are not enough rows available in the Excel satellite range, information is returned to SAP and an error message is delivered there (processing is aborted).
- A similar function can be implemented with a constant for MOD applications. In this case, a message appears immediately after the list has been compiled in SAP, meaning that Excel is not called at all.

### 8.7 Formulas and performance in the satellite range (KeyPointer “FORMULA”)

Satellite ranges are often defined with a fixed length in the Master (maximum length according to the expected data); formulas allow additional calculations of values. If these formulas are distributed over the entire satellite range, they are also executed in rows that do not contain any data. This can have an influence on the performance if there are large satellite ranges (e.g. when reading reference data).

On the other hand, the row structure is often the same throughout: it is therefore advisable to store formulas centrally and only transfer them to the rows containing data after reading from SAP.

For this, the KeyPointer FORMULA can be entered in the column axis of the satellite. The selected cells can then be provided with formulas to copy them into the respective fill area of the field.

What happens if this KeyPointer exists in the satellite: Before reading reference data from SAP, all previous formulas are removed from the satellite range and after reading - on the basis of FORMULA - are reinserted into the rows with data.

The procedure is intended for all satellites used in the Master. It can also be used if a structured table is located above the satellite range.

|              |  |
|--------------|--|
| <b>Note:</b> | In this procedure, formulas are only stored in the rows that have been filled with data from SAP. Consequently, inserting new rows must be handled individually during planning. |
|--------------|--|

### 8.8 Using “structured tables” with satellites

In newer versions, Excel provides a special function to make the processing of defined table ranges easier: The Excel keyboard shortcut Ctrl+T defines such ranges and they are then labeled as “structured tables”. Some advantages:

- The contents of the header ranges are interpreted as fields of the table.
- The table expands automatically as soon as values are entered directly beneath, to the left or to the right.

Allevo also supports these functions (with restrictions) for satellite ranges; formulas may affect performance (for large amounts of data).

|              |  |
|--------------|--|
| <b>Note:</b> | With “structured tables” formulas in one cell are automatically used for all cells in the column. There are cases in which this “AutoFill” function is not desirable (e.g. for performance reasons when reading a large number of satellite rows). In Excel, the settings can be made for the respective workstation (see corresponding Excel settings). |
|--------------|--|

The parameter should be reset to the original entry when leaving Allevo. For this, there are two Allevo macros:

- *DeactivateAutoFillFormulas* deactivates the AutoFill function for formulas; the macro should be entered for the events WB\_OBEN or BER\_READ.
- *ResetAutoFillFormulas* resets the Excel settings to the previous status in the workstation (the macro should be entered for WB\_CLOSE).

## 9 Navigation – the NavigationCorner and the Menu ribbon

Navigation within the Allevo Master is completely controlled using commands in the Navigation Menu ribbon, i.e. by clicking one of the symbols, the user can navigate to special views of the planning form.

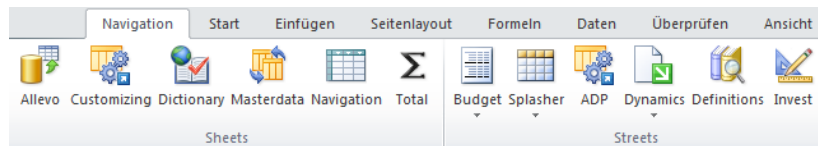


Figure 9.1: Example of a menu ribbon with different elements

Each element or symbol in the menu ribbon corresponds to a so-called NavigationCorner. This navigation corner controls the view to a particular “Street”. In particular, rows (row filters) and columns (column filters) are shown or hidden, but many additional configurations of the view are also made.

### 9.1 Structure of a NavigationCorner

Figure 9.3 shows the structure of a NavigationCorner. The Navigation corner is made up of the two axes NavigationXXRow and NavigationXXColumn. In Figure 9.3, the row axis is marked and the Excel name Navigation01Row is shown in the upper left hand corner.

Entering this Corner “Navigation01” into the “NavigationViews” table makes the button available in the “Start” tab of the Excel ribbon. In this case, a view ID Main01 has been used in the table and there are other, different view definitions under additional view IDs (Main02, and so on).

| SheetName | View   | NavigationCorner | ViewName      | Tooltip                  |
|-----------|--------|------------------|---------------|--------------------------|
| Allevo    | Main01 | Navigation01     | Start         | Start your planning here |
| Allevo    | Main02 | Navigation01     | Cost Elements |                          |

Figure 9.2: NavigationViews table

Depending on the view, the following settings can be configured:

- Activate certain rows using a custom row filter (not a standard Excel filter)
- Activate certain columns using a custom column filter (not a standard Excel filter)
- Fix the window: this is done to the left above the letter combination (here: M) found by the FREEZE-KeyPointer in column and row
- Setting the Excel view zoom level
- Setting pre-defined column width and row height (formulas are possible)
- If necessary, execution of a stored macro

- Define a custom icon for each view (custom icons are supported for version 3.5 and higher)
- Store Tooltips and urls

| Navigation01Row |                   |         |            |        | AF       | AG | AH | AI | AJ | AK | AL | AM |
|-----------------|-------------------|---------|------------|--------|----------|----|----|----|----|----|----|----|
| 1               | Navigation01      |         |            |        |          |    |    |    |    |    |    |    |
| 2               | O1                | SHOWROW | OWROWFORMU | FREEZE | HEIGHT01 |    |    |    | D  |    |    | R  |
| 3               | SHOWCOLUMN        |         |            |        |          | *  | *  |    |    |    |    |    |
| 6               | SHOWCOLUMNFORMULA |         |            |        |          |    |    |    | A  |    |    |    |
| 7               | FREEZE            |         |            |        |          |    |    |    |    |    |    |    |
| 8               | WIDTH01           |         |            |        |          | 1  | 11 | 25 | 11 | 4  | 4  | 11 |
| 31              |                   |         |            |        |          |    |    |    |    |    |    |    |
| 32              |                   | 1       |            |        |          |    |    |    |    |    |    |    |
| 33              |                   | 1       |            |        |          |    |    |    |    |    |    |    |
| 35              |                   | 1       |            | M      |          |    |    |    |    |    |    |    |
| 36              |                   | 1       |            |        |          |    |    |    |    |    |    |    |
| 37              |                   | 1       |            |        |          |    |    |    |    |    |    |    |
| 38              |                   | 0       |            |        |          |    |    |    |    |    |    |    |
| 39              |                   | 0       |            |        |          |    |    |    |    |    |    |    |
| 41              |                   | 1       |            |        |          |    |    |    |    |    |    |    |
| 42              |                   | 0       |            |        |          |    |    |    |    |    |    |    |
| 43              |                   | 0       |            |        |          |    |    |    |    |    |    |    |
| 44              |                   | 0       |            |        |          |    |    |    |    |    |    |    |

| 1000                   | Kantine                     | EC | ADP  | Sender |
|------------------------|-----------------------------|----|------|--------|
|                        | Hertwig                     |    | Year | Month  |
| Statistical Key Figure |                             |    |      |        |
| 2010                   | Square meter                |    | A    |        |
| 4510                   | Power usage, drying process |    | S    |        |
| Activity Type          |                             |    |      |        |
| 1520                   | # Mittagessen               |    |      | 1      |

Figure 9.3: NavigationCorner

Similar to Excel, rows are usually provided with numbers and columns with letters, but this is not a technical limitation.

In addition to the view definition described above, there are two additional tables for creating ribbons per worksheet (table: NavigationSheets), and storing macros via ribbon entry (table: NavigationCustomButton). These tables are explained in more detail in the Excel Master on the navigation sheet.

|              |  |
|--------------|--|
| <b>Note:</b> | You can insert as many additional help rows and columns as you like.   |
|              | The navigation functions on the union of the ranges covered by both axes. Outside this range, the visibility is not changed.   |
|              | While the control column of the row filter uses a simple CONCATENATE formula, the control column of the column filter uses an Allevo-specific formula. This returns a FALSE value if there is a different value in the cell after clicking the button than the value defined for the button. Cells with FALSE contents are hidden. |

## 9.2 Memory setting of a NavigationCorner

The previously executed navigation and its most important settings can be queried via formula from the NavigationHistory table. The most recently executed navigation is saved in the column “Current”, the one before that in the “Last” column.

**History**

| View Status  |            |            |
|--------------|------------|------------|
| Property     | Current    | Last       |
| View         | Invest01   | Main01     |
| FilterRow    | MainRow    | MainRow    |
| FilterColumn | MainColumn | MainColumn |
| ShowRow      |            | 1          |
| ShowColumn   | I          | A          |
| Freeze       | I          | M          |
| Cell         | \$AG\$285  | \$BQ\$25   |

**Figure 9.4: NavigationHistory: Finding the most recently executed navigation**

**Note:** The reference to the last executed view is useful, for example, for the following constellation: Button A selects row filter 1 (FilterRow = 1), button B selects row filter 2 (FilterRow = 2). If Button C is used, it should keep the row filter of the previous button, regardless of whether Button A (i.e. 1) or Button B (i.e. 2) was used.

**9.3 Context menu**

As of Excel version 2010, all entries from the menu ribbon are available in the context menu (right mouse click).

The context menu contains the same entries and icons as the menu ribbon and can also execute the same macros and callers.

The context menu can also be inserted into existing Allevo Masters if they already work with the navigation described above. Please contact our support team for further help with this issue ([support@kern.ag](mailto:support@kern.ag)).



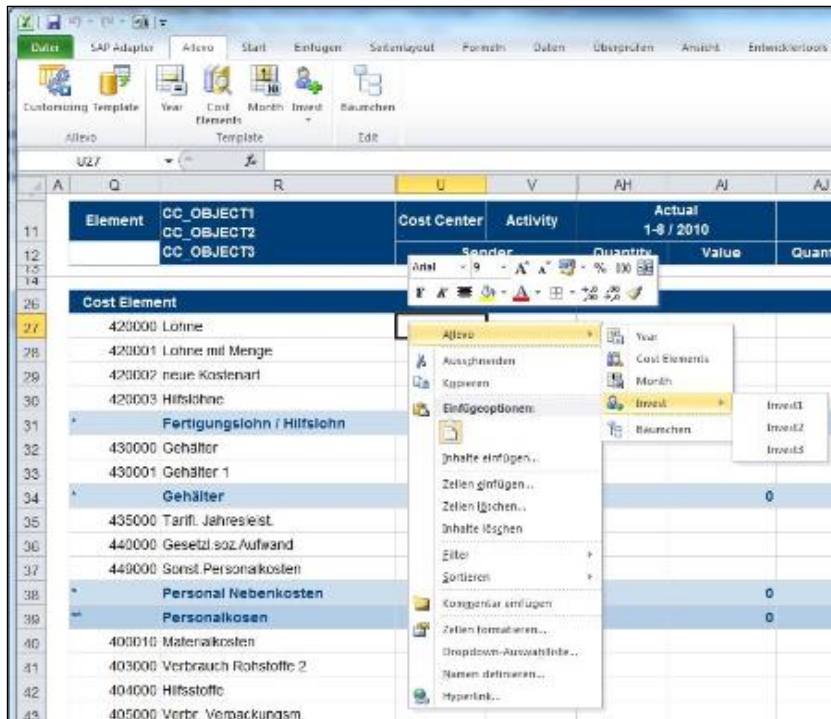


Figure 9.5: Context menu in the Allevo Master

### 9.4 Toggle button for showing and hiding zero rows (optional)

The toggle button “SuppressZeroRows” is an extension to the navigation and either shows or hides the zero rows, eliminating the need for two separate buttons for this task. One prerequisite is that the NavigationCorner contains the corresponding formulas that can access a flag in the Customizing sheet.

The NavigationCorner must be stored with the corresponding formulas in order to access this feature. The formula accesses a flag that is read and reset by the macro ShowHideZeroRows. In the example below, the range is called Z\_SUPPRESS\_ZERO, but can be chosen freely.

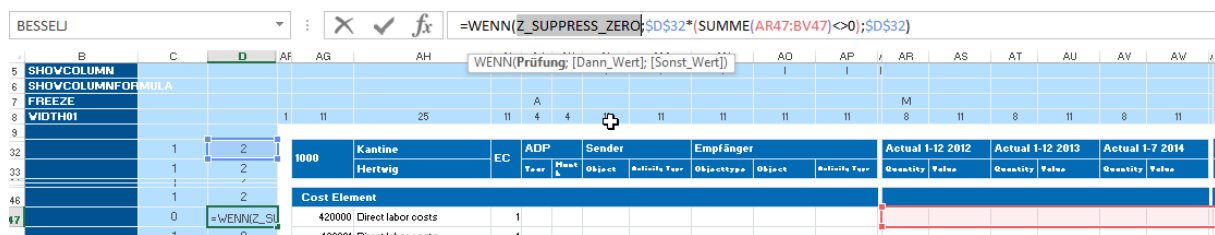


Figure 9.6: Macro ShowHideZeroRows

In the *Customizing* sheet, the required flag is created with the ID *SuppressZeroRows* and the initial value *False* in the *Common Flags* range. The flag is assigned the range name that is to be accessed via formula (see above).

Z\_SUPPRESS\_ZERO

|     | G                | H     |
|-----|------------------|-------|
| 141 | Common Flags     |       |
| 142 |                  |       |
| 143 | 20               | 20    |
| 144 |                  |       |
| 145 | ID               | Flag  |
| 146 | SuppressZeroRows | False |
| 147 |                  |       |

Figure 9.7: Common Flags range

The toggle button is created just as you would create a custom button, a filter would be a suitable icon here, but as always it is freely selectable. The macro ShowHideZeroRows is entered here. Since the identifier is to change at runtime, depending on the status of the toggle button, it is also assigned the range Z\_SUPPRESS\_ZERO\_CAPTION.

Z\_SUPPRESS\_ZERO\_CAPTION

Show Zero Rows

|    | AE            | AF   | AG                 | AH               | AI             | AJ                  |
|----|---------------|------|--------------------|------------------|----------------|---------------------|
| 18 | Custom Button |      |                    |                  |                |                     |
| 20 | Custom Button |      |                    |                  |                |                     |
| 21 | SheetName     | View | Icon               | MacroName        | Caption        | Tooltip             |
| 22 | Total         |      | Printer            | PrintStreet      | Print          | Send to Printer     |
| 23 | Allevo        |      | Printer            | PrintStreet      | Print          | Send to Printer     |
| 24 | Allevo        |      | DetailsHouseDollar | ShowHideZeroRows | Show Zero Rows | Show/Hide Zero rows |

Figure 9.8: Custom Button range

The toggle button should change its freely selectable and translatable identifier depending on its status. To make this possible, the FormCaptions range can be manipulated. Instead of a form, a ContainerID SuppressZeroRows with the fake labels lblShowRows and lblHideRows is entered. You can then enter freely selectable identifiers for these. If no entries are found here, the default names "Show Zero Rows" and "Hide Zero Rows" are used. Depending on the status, the identifier is copied to the range Z\_SUPPRESS\_ZERO\_CAPTION.

Forms and Controls

| FormName         | ControlName | ControlCaption |
|------------------|-------------|----------------|
| SuppressZeroRows | lblShowRows | Show Zero Rows |
| SuppressZeroRows | lblHideRows | Hide Zero Rows |
|                  |             |                |

Figure 9.9: "Forms and Controls" range

## 10 Additional settings on the |CUSTOMIZING| sheet

### 10.1 Settings, sheet and workspace protection

Most settings are usually very specific and need to be adjusted/set only once, if at all.

#### Settings

| SettingKey                                   | LogicalValue | Description   | LogicalDefault | Custom |
|--|--------------|---|----------------|--------|
| <b>Sheet protection</b>                      |              |   |                |        |
| Password                                     |              | No sheet and workbook protection if empty             |                |        |
| Master adjustment after opening in SAP/Excel |              |   |                |        |
| ShowCustSheet                                | FALSCH       | Hide this sheet always on Allevo startup              | FALSCH         |        |
| ShowRibbon                                   | WAHR         | Show Allevo ribbon                                    | WAHR           |        |
| ShowFormBar                                  | WAHR         | Show Form Bar   | WAHR           |        |
| <b>Settings for Multi and MultiObject</b>    |              |   |                |        |
| MOMDirect                                    | WAHR         | MOM without assigned objects                          | WAHR           |        |
| NamingRule                                   |              | Naming rule for Multi Page                            |                |        |
| CopyMultiSheet                               | FALSCH       | Protect digital signature when creating multi sheets  | FALSCH         |        |
| UseMOMObjectType                             | WAHR         | Use object type                                       | WAHR           |        |
| <b>Settings for data format in Excel</b>     |              |   |                |        |
| FirstSheet                                   |              | Name of first sheet                                   |                |        |
| DivValue                                     | FALSCH       | Dividing values by 1000                               | FALSCH         |        |
| DivQuantity                                  | FALSCH       | Dividing quantities by 1000                           | FALSCH         |        |
| ShowDelWarning                               |              | Text for warning message when using the delete button |                |        |
| WriteEmptyComments                           | FALSCH       | Insert dummy character when deleting a comment        | FALSCH         |        |
| StatKey                                      | ^{s}         | Select a shortcut key                                 |                | ^{s}   |
| PickListKey                                  | ^{c}         | Select a shortcut key                                 |                | ^{c}   |

Figure 10.1: Settings in the Allevo Customizing sheet

Notes on individual settings:

- An entry in "Password" protects the sheets and workbook as soon as the Master is called Inplace or via the ABC (entry in the orange marked cell). The protection also applies in offline mode, depending on whether or not the read objects are recognized. This recognition is carried out by means of an entry >0 in the Status column of the object-related table ZZLocalInformation on the sheet "Infos".
- "ShowCustSheet": if set (True), the Customizing sheet is hidden when the workbook is opened.
- "ShowRibbon" must be set (True) to show the Allevo ribbon when opening the workbook.
- "ShowFormBar" controls the display of the Excel toolbar. The entry is evaluated when the workbook is opened in Inplace or via ABC.
- "MOMDirect" specifies how an Allevo Master is to be used in MultiObject mode: for planning in list form (True) or with variable sheet assignment (False). See overview in Section 14.
- NamingRule defines a naming rule for the sheets in MultiPage or MultiObject-mode with variable sheet assignment. Without specific name assignment, the standard specifications (i.e. "IPP\_xxxx") apply.
- "CopyMultiSheet": In Excel environments with high security settings, the Allevo Master is usually provided with a code signature. Due to Excel-specific properties, this signature can be lost if Allevo tries to create the required sheets in MultiPage mode. In this case, copying should be done using the "CopyMultiSheet" method - "True" should be set for this setting.  
The method cannot be used if satellites are created as structured tables on the sheet.

- If a sheet name is entered in “FirstSheet”, this sheet is activated when opening and after reading/planning.
- “DivValue”: yearly and monthly values are divided by 1000 when reading and multiplied by 1000 when planning.
- “DivQuantity” activates the same function as “DivValue” for yearly and monthly quantities (applies only to balloon variant, if the suffix ‘\_Q’ is stored for column definition).
- “ShowDelWarning” defines a text, which is always output when the contents of a cell have been deleted (a zero is simultaneously entered into the cell). With Allevo 3.4 this function has been replaced by “WriteTechnicalZero”.
- “WriteTechnicalZero” activates a function to automatically enter a 0 in the respective cell when deleting cell contents (via keys like Del or Backspace) or for all empty cells in certain ranges. This transfers the cell content to SAP and also resets the planned value. Optionally, a confirmation can be requested from the planner (PopUp before setting the zero values).
- “WriteEmptyComments” solves a special situation when deleting texts in a comment column. If active (True), the special character “~” is entered in the respective cell (otherwise the empty cell would not be transferred to SAP and the original comment would remain).
- “LockShowAll” suppresses the display of control information for rows and columns in the Allevo Master (key combination “Ctrl+Shift+A” may no longer be possible). Especially recommended for sensitive data.
- FISH (Fishbase=Fixsheet)  
MultiObject for profit center: Master with fixed sheets for different company codes with the same object.
- PickList (optionally available) allows two-column specifications for drop-down lists in Excel; the appropriate structured table should be created for a sheet without navigation; structured table: PickList<Table>.
- StatKey (optionally available) displays the statistical value of a selected range: structured table: ZZKITStatisticBase.

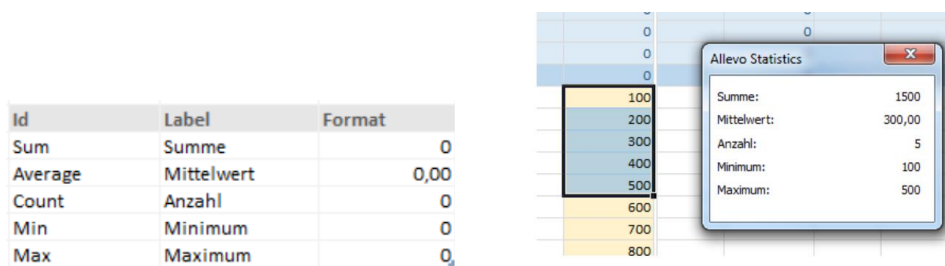


Figure 10.2: Allevo statistics

**Note:**

Occasionally, when opening the Master in Excel 2007/2010, problems can occur if workbook protection is set. This is primarily due to security settings that cannot be controlled via Allevo.

Allevo tries to avoid this issue, however, by removing the workbook protection when closing the folder and resetting it at a later time.

### 10.2 Headline – Column headers

The Allevo Master is created in such a way that all settings for periods, years, versions, etc., to be executed on the SAP side of Allevo can be automatically shown in Excel. In particular, this applies to the column headings.

To do this, the column definitions are transferred from SAP to Excel each time the Allevo Master is opened in SAP and stored temporarily in the Customizing sheet (see table *ZZCustomizingHeadlineBase*).

| HeadlineDescription | HeadlineRange | ValueCategory | ValueCategoryDes | VersionFrom | Mc | MonthDesFrom | YearFrom | VersionTo | MonthTo | MonthDesTo | YearTo | HeadlineText |
|---------------------|---------------|---------------|------------------|-------------|----|--------------|----------|-----------|---------|------------|--------|--------------|
| Actual 1-12 2014    | CM_R1         | 1             | Actual           | 0           | 1  | Jan          | 2014     | 0         | 12      | Dec        | 2014   |              |
| Plan 1-12 2015      | CM_R2         | 2             | Plan             | 3           | 1  | Jan          | 2015     | 3         | 12      | Dec        | 2015   |              |
| Plan 1-12 2016      | CM_RB1        | 2             | Plan             | 3           | 1  | Jan          | 2016     | 3         | 12      | Dec        | 2016   |              |
| Plan 1-12 2017      | CM_RB2        | 2             | Plan             | 3           | 1  | Jan          | 2017     | 3         | 12      | Dec        | 2017   |              |
| Actual 1-12 2013    | CM_RH1        | 1             | Actual           | 0           | 1  | Jan          | 2013     | 0         | 12      | Dec        | 2013   |              |
| Actual 1-12 2012    | CM_RH2        | 1             | Actual           | 0           | 1  | Jan          | 2012     | 0         | 12      | Dec        | 2012   |              |
| FC 1-12 2014        | CM_W1         | 2             | FC               | FC1         | 1  | Jan          | 2014     | FC1       | 12      | Dec        | 2014   |              |
| Plan 1-12 2015      | CM_W2         | 2             | Plan             | 3           | 1  | Jan          | 2015     | 3         | 12      | Dec        | 2015   |              |
| Actual 1-12 2014    | CX_RR         | 1             | Actual           | 0           | 1  | Jan          | 2014     | 0         | 12      | Dec        | 2014   |              |
| Plan 1-12 2015      | CX_WW         | 2             | Plan             | 3           | 1  | Jan          | 2015     | 3         | 12      | Dec        | 2015   |              |

Figure 10.3: Assembling the headers in the |Headline| range of the |CUSTOMIZING|-sheet

With this information from SAP, a heading is created in the column "HeadlineDescription" using an Excel formula. The cells in this column are given an Excel cell name that corresponds to the ID of the column definition of the respective SAP column definition (in Figure 10.3, for example, "CX\_RR"). This name can now be simply transferred to the cell, which should then finally contain the heading.

**Caution:** Since the position of the row (e.g. for CX\_RR) is dependent on which column definitions are active (CM\_R1 comes before CX\_RR), the cell names are newly generated each time the file is opened in SAP. A cell definition in the Master may be changed when opening SAP.

The language dependency of the headings can be automated in conjunction with the |DICTIONARY| (see Chapter 11).

| Element             | Actual 1-12 / 2009 | Actual 1-8 / 2010 | Plan V0 1-12 / 2010 |
|---------------------|--------------------|-------------------|---------------------|
| CC_OBJECT1          | Quantity           | Quantity          | Quantity            |
| CC_OBJECT2          | Value              | Value             | Value               |
| CC_OBJECT3          |                    |                   |                     |
| <b>Cost Element</b> |                    |                   |                     |
| 420000 Wages        |                    |                   |                     |
| 420001 Overtime     |                    |                   |                     |
| 420002 Benefits     |                    |                   |                     |

Figure 10.4: Use as column header

Optionally, the characteristics of a column definition can also be changed directly by the user, e.g. to temporarily read data from another year (this function is described in Section 18.2).

### 10.3 UserExits – automatic calling of macros and Callers (views)

Under |User Exit|, the Allevo Master provides a list of “events” for which macros and Callers can be entered.

**User Exit**

| Basic          |        | Macro              |                 |
|----------------|--------|--------------------|-----------------|
| Event          | View   | Macro1             | Macro2          |
| AFT_LEAD_SAT   |        |                    |                 |
| AFT_PLAN       | Main01 |                    |                 |
| AFT_PLAN_S     |        |                    |                 |
| AFT_READ       | Main01 | SplasherActivate   |                 |
| AFT_READ_S     | Main01 |                    |                 |
| AFT_SHEET_GEN  |        |                    |                 |
| AFT_SHEETS_GEN |        |                    |                 |
| BEF_PLAN       |        | SplasherDeactivate |                 |
| BEF_PLAN_S     |        |                    |                 |
| BEF_READ       |        | SplasherDeactivate | TranslateSheets |
| BEF_READ_S     |        |                    |                 |

Figure 10.5: Calling the Navigation (caller) or a macro depending on “events”

In Figure 10.5 above, it is specified that after reading the data from SAP (AFT\_READ) the caller "Main01" is called. This means that a certain view is created after reading, for example, hiding zero lines and displaying certain columns.

In addition, the macro "TranslateSheets" was executed before opening the file in SAP (OPEN\_IN\_SAP).

The events provided by Allevo have the following meanings and call times:

|   |   |
|---|---|
| <b>AFT_LEAD_SAT</b>                           | Relevant for satellites for which constant READ_ORDER_SAT is active: is executed AFTER a satellite has been read, but BEFORE reading the plan data (i.e. between BEF_READ and AFT_READ). The satellite data can thus serve as the basis for further reading functions (sequence applies per sheet in MultiPage mode). |
| <b>AFT_PLAN</b><br><b>AFT_PLAN_S</b>          | After transferring the plan data to SAP once at the end (with the ending _S after each worksheet)   |
| <b>AFT_READ</b><br><b>AFT_READ_S</b>          | After reading the plan data from SAP once at the end (with the ending _S after each worksheet)  |
| <b>AFT_SHEETS_GEN</b><br><b>AFT_SHEET_GEN</b> | After each generation of MultiPage worksheets once at the end (for SHEET after each worksheet for Excel internally)   |
| <b>AFT_START_SAT</b>                          | After reading all start satellites / after transferring the MOD satellites (and thus also after SAT00)  |
| <b>AFT_TT</b>                                 | To refresh the view of the totals sheet, only works with the corresponding add-on module 'TotalSheet'.  |
| <b>BEF_PLAN</b><br><b>BEF_PLAN_S</b>          | Before the transfer of plan data, once at the beginning. (with _S after each worksheet)   |
| <b>BEF_READ</b><br><b>BEF_READ_S</b>          | Before reading plan data from SAP completely. (with _S after each worksheet)  |
| <b>MOD_TEMPLATE</b>                           | Between OPEN_IN_SAP and AFT_SHEET_GEN, before setting the header data.  |
| <b>OPEN_IN SAP</b>                            | Opening the Master in SAP: Executed when the Master is started in the planning environment (SAP Inplace or via ABC). Called only once.  |

|   |  |
|---|--|
| <b>WB_OPEN</b>                            | Opening the Master outside of SAP  |
| <b>WB_CLOSE</b>                           | Close Master outside of SAP  |
| <b>BEF_RD_SAT</b>                         | Before reading all satellites;<br>See also the additional function "Read satellite" and "FP-Read".                     |
| <b>AFT_WR_SAT</b>                         | After saving satellite data in SAP;<br>See also the additional function "Save satellite".                              |
| <b>AFT_RD_SAT</b>                         | After reading satellite data in SAP  |
| <b>BEF_WR_SAT</b>                         | Before saving satellite date in SAP  |
| <b>BEF_RD_SAT</b>                         | Before reading satellite data in SAP   |
| <b>BEF_ORDS00</b><br><b>...BEF_ORDS99</b> | Before reading a satellite again (one event per satellite);<br>Called only after opening an Allevo file saved offline. |

The "SHUTTLE"-events are only executed when using the Master via the Shuttle-transaction (see Shuttle manual for more information).

#### 10.4 Global Information

Global information (KernGlobal parameters) are parameters that can be used to transfer further information from the planning workflow to the Allevo Master. They are therefore a supplement to the object-related information (KernLocal parameters).

In contrast to object-related information, global information is not directly related to the planning object, but can also arise from other aspects of the planning workflow. Examples of global information include the planning layout and the user.

Both object-related and global information are stored in the Infos table sheet. The following list shows currently available information:

| ID Global Info                           | Contents  | Field type | Length |
|--|---|------------|--------|
| GLOBAL_ACTUALYEAR                        | Year from CX_RR (or current calendar year if CX_RR is not set).   | NUMC       | 4      |
| GLOBAL_ADP                               | Contents of the constant LSTAR_VARIATOR   | CHAR       | 1      |
| GLOBAL_COMPANYCODE                       | Company code of the start object  | CHAR       | 50     |
| GLOBAL_CONTROLLINGAREA                   | Active user controlling area (from the start screen)  | CHAR       | 4      |
| GLOBAL_COSTCENTER                        | Cost center for the representative element (OR: AUFK-KOSTV, PR: PRPS-KOSTL, otherwise empty)                                    | CHAR       | 10     |
| GLOBAL_CURRCONTROLLINGAREA               | ISO code of the controlling area currency   | CHAR       | 2      |
| GLOBAL_CURROBJECT                        | ISO code of the object currency   | CHAR       | 2      |
| GLOBAL_CUSTINFO1 to<br>GLOBAL_CUSTINFO10 | Additional information on master data based on the Allevo constant MAP_FIELDxx<br>(see documentation on constant for more info) | CHAR       | 50     |
| GLOBAL_DATE                              | Current date in the format YYYYMMDD   | DATS       | 8      |
| GLOBAL_FCODE                             | Function code used to start Allevo, for example: ONLI, ONLIREP, FOPEN   | CHAR       | 70     |
| GLOBAL_FROM                              | MultiPage: "Value from" from the start screen, MOD: representative element of the start group, otherwise empty                  | CHAR       | 12     |

| ID Global Info        | Contents   | Field type | Length |
|-----------------------|--|------------|--------|
| GLOBAL_GROUP          | Group that is assigned as a hierarchy area in the object master record (only for KS, PC, GP, KX)   | CHAR       | 12     |
| GLOBAL_ISOLANGUAGE    | SAP language key (2-character key)   | CHAR       | 2      |
| GLOBAL_LANGUAGE       | Login language in SAP 1-character (e.g. "D", "E")  | CHAR       | 1      |
| GLOBAL_LAYOUT         | ID code of the current Allevo layout   | CHAR       | 4      |
| GLOBAL_LOGSYS         | Logical system of the login client   | CHAR       | 32     |
| GLOBAL_LSTARS         | Only for object type KS: X if constant LSTAR_FROM_SAP is active, otherwise empty.  | CHAR       | 1      |
| GLOBAL_OBJECT         | Relevant object with the following contents: <ul style="list-style-type: none"> <li>• Single and Standard MO: start object,</li> <li>• MultiPage, Tree and MOD: when starting with a group, this is the first object of the group (representative object)</li> <li>• MultiPage and MOD: if starting via multiple selection or the "from-to" range, this is the first object in the list (representative object)</li> </ul> | CHAR       | 24     |
| GLOBAL_OBJECTGROUP    | Id code of a group: <ul style="list-style-type: none"> <li>• with MultiPage, MOD or calling via Tree, this is the group in the start screen;</li> <li>• when starting via single transaction, this is the group from the 1:n assignment (otherwise empty).</li> </ul>  | CHAR       | 15     |
| GLOBAL_OBJECTTYPE     | Object type  | CHAR       | 2      |
| GLOBAL_PERITO         | Period to for the current year (from column definition CX_RR or "12" if there is no entry for CX_RR).  | NUMC       | 3      |
| GLOBAL_PLANYEAR       | Plan year (from the start screen)  | NUMC       | 4      |
| GLOBAL_PROFITCENTER   | Assigned profit center for KS, OR, PR and BP, otherwise empty  | CHAR       | 10     |
| GLOBAL_PROJECT        | Start project (only in PSP-Allevo)   | CHAR       | 24     |
| GLOBAL_READ           | This code is derived from VBA (not transferred from SAP)   | -          | -      |
| GLOBAL_REPRESENTATIVE | Representative element when entering via a group (Multi / MOD)   | CHAR       | 24     |
| GLOBAL_RESPONSIBLE    | Responsible person of the representative object (KS: CSKS-VERAK, OR: AUFK-USER2, PC: CEPC-VERAK, PR: PRPS-VERNA)   | CHAR       | 25     |
| GLOBAL_SETCLASS       | Setclass of the start object   | CHAR       | 4      |
| GLOBAL_SHEETCOMMENT   | Sheet comment  | CHAR       | 5000   |
| GLOBAL_STATUS         | Status of the relevant object  | NUMC       | 2      |
| GLOBAL_SYSID          | SAP System-ID (e.g. D46, P01 etc.)   | CHAR       | 8      |
| GLOBAL_TEXT           | (Short) text of the representative object (KS: CSKT-KTEXT, OR: AUFK-KTEXT, PC: CEPCT-KTEXT, PR: PRPS-POST1)  | CHAR       | 40     |
| GLOBAL_TO             | MultiPage & MOD: "Value to" from the start screen, otherwise empty   | CHAR       | 12     |
| GLOBAL_TRANSACTION    | Transaction code of the SAP transaction used to start, e.g. /KERN/IPPKS, /KERNIPPORMO, ...   | CHAR       | 50     |



| ID Global Info | Contents   | Field type | Length |
|----------------|--|------------|--------|
| GLOBAL_USER    | SAP name of the current user<br>(see also constant USER_DATA). | CHAR       | 12     |
| GLOBAL_VERSION | Plan version (from the start screen)                           | CHAR       | 3      |

**Note:** Some of the information relates to properties of an object; in such cases they are related to the so-called "representative object" used to start an Allevo transaction.

## 10.5 Captions form

The Captions form can be used to manage and translate identifiers for formulas and controls.

Formulas are being used more and more frequently with Allevo. To maintain the freedom to change or translate identifiers at short notice, the controls can be managed in the FormCaptions range. The labels are read with the VBA function GetFormCaption. If no identifiers are entered, the form uses a default identifier.

**Note:** The use of this function is always related to macros and should not be done without consultation!

## 10.6 Common Flags

In the Common Flags you can manage customer specific and other flags.

In column one, the identifier for the flag is entered, in column two the value. In VBA, this can be accessed using the standard functions GetCommon FlagByID and SetCommon FlagByID.

**Note:** The use of this function is always related to macros and should not be done without consultation!

## 10.7 Splasher / distribution

Allevo has functions to convert yearly data into the corresponding monthly values and vice versa (horizontal distribution). Alternatively, you can also use "vertical distribution" to automatically distribute inputs at the group level to the corresponding cost components. The reference values required for distribution can be defined in the Master.

For the Standard distribution, the following Customizing settings are relevant.

**SplasherCommonSettings**

| Common            | Setting |
|-------------------|---------|
| activationByStatu | 0       |
| horizontal        | 1       |
| vertical          | 0       |
| sumFlag           | 1       |
| decimals          | 2       |
| writeZero         | 1       |
| AllSumLevels      | 1       |
| CheckSettings     | 1       |

Figure 10.6: Parameters on the left, the cell with the corresponding value on the right

## Meaning:

- acitvationByStatus: if true the Splasher is only active if the CC\_STATUS > 0.
- horizontal: activate horizontal distribution
- vertical: activate vertical distribution
- sumFlag: Sum Flag (only relevant if vertical = true)
- decimals: rounding to x decimal places
- writeZero: help function to write a 0 when pressing the delete key
- AllSumlevels: execution only for sum levels >1 for vertical distribution
- CheckSettings: test for inconsistent setup of the Splasher

|              |  |
|--------------|--|
| <b>Note:</b> | The Splasher is usually individually tailored to customer requirements in the implementation project and should not be carried out without consultation. |
|--------------|--|

## 11 The Dictionary

### 11.1 Activating the Dictionary

The Dictionary function has been created to enable you to translate any text entries depending on the login language. For this purpose, the translations are stored centrally in the Master, namely on the |DICTIONARY| worksheet.

**Note:** To activate the Dictionary, the function "TranslateSheets" must be entered in the |CUSTOMIZING| sheet, under |User Exit|, for an appropriate event.

#### User Exit

| Basic       |        | Macro           |        |
|-------------|--------|-----------------|--------|
| Event       | View   | Macro1          | Macro2 |
| OPEN_IN_SAP | Main01 | TranslateSheets |        |

Figure 11.1: Activation of the Dictionary under the event OPEN\_IN\_SAP

### 11.2 Functionality of the Dictionary

All text entries of the Master and the corresponding translations for the desired languages are centrally entered and managed in a list on the |DICTIONARY| sheet. The list can be maintained manually in the Master or can be filled via satellite 0 when Allevo is launched.

#### Constants per language for Translation

| Index | E                               | D                                   |
|-------|---------------------------------|-------------------------------------|
| 1     | Activity Dependend Planning     | leistungsarten-abhängige Planung    |
| 2     | Activity independend            | leistungsunabhängig                 |
| 3     | Activity Input (Partner Object) | Leistungsaufnahme (Ursprungsobjekt) |
| 4     | Activity Input (Source Object)  | Leistungsaufnahme (Partnerobjekt)   |
| 5     | Activity Output                 | Leistungsabgabe                     |
| 6     | Activity Type                   | Leistungsart                        |
| 7     | Actual                          | Ist                                 |

Figure 11.2: Dictionary entries

When the "TranslateSheets" function is triggered, the texts of those cells in the workbook that have the format template "KernHeader1... n" or "KernHeadline1... n" are translated. The translation is done into the language that was transferred to the Excel file as the login language in SAP's global parameter Language.

The translation is carried out if the entry in the correspondingly formatted cell in the Dictionary has an entry in the column "Language ID". If a cell in the Master contains the entry LART, this entry is translated into either "Activity" or "Leistung" depending on the language in the example in the upper figure.

The following rules apply:

- Entries created with a formula are not translated

- If there is no translation for an entry in the Dictionary column “Language ID”, then the entry is not translated or overwritten.
- If a language from the global parameter Language is not found in the dictionary, the translation proceeds in the Standard language (this is English in the standard Master). If this is also not found, translation is not executed.

|              |  |
|--------------|--|
| <b>Note:</b> | For headers and titles that are NOT to be translated, the format templates “KernTop1” to “KernTop3” can be used. |
|--------------|--|

## 12 The totals sheet and the summary sheet

### 12.1 Simple totals sheet (SUM)

Users of Allevo Multi often request to extend the Allevo Master by a summary sheet recording the totals per cost element etc. for all cost centers (orders/WBS element) in the file.

The summary sheet uses the Excel *SUM* formula that not only sums rows or columns but also totals individual cells of a predefined sequence of sheets. The following steps are used to set up the summary sheet:

1. Adding a copy of the template sheet
2. Creating an empty spreadsheet and classifying spreadsheets as shown in Figure 12.1.

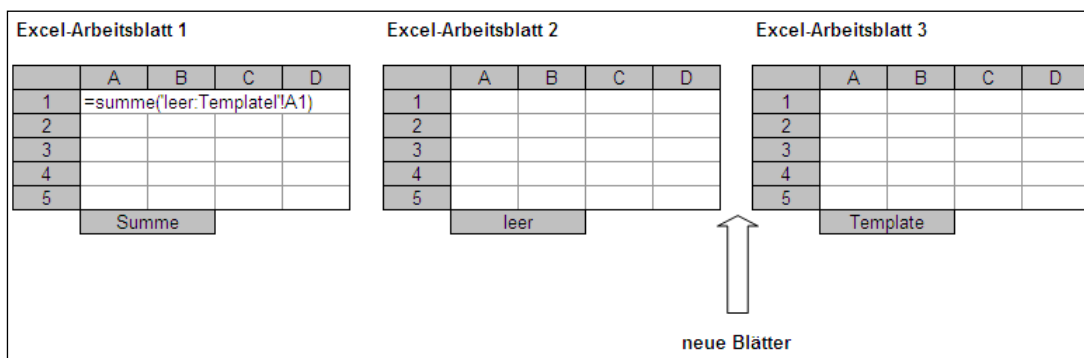


Figure 12.1: How the summary sheet works

1. Inserting the formula in the totals sheet, which now draws the sum of a cell from the empty sheet to the |ALLEVO| sheet (e. g. "`=SUM('empty:Allevo'!A1)`"). When new worksheets are added, they will be inserted between the empty and the |ALLEVO|-sheet and thus automatically become part of the total. The blank sheet serves only to avoid circuitous references in the sum formula and can be hidden if desired.
2. Format the totals sheet according to your own requirements. As a rule, the comment column as well as the satellite ranges on the summary sheet can be deleted, since experience has shown that they are not used for totals calculation.

|              |   |
|--------------|---|
| <b>Note:</b> | Once the totals have been inserted in the totals sheet, you can also delete columns and rows in the totals sheet. However, if the column or row structure in the  ALLEVO  sheet is changed, the totals sheet must be adjusted from the respective position onwards. |
|              | When copying the  ALLEVO  sheet, its name ranges are also copied. Although this does not affect the ability of Allevo to run, you should delete the names ROW_Y, ROW_M, COL_Y and COL_M on the totals sheet.  |
|              | When copying the  ALLEVO  sheet, the navigation function is also transferred. It should be adjusted for the totals sheet, if necessary.   |

### 12.2 Macro based totals sheet (Overview summary sheet)

In principle, the macro-based total sheet corresponds to a classic operational accounting sheet: the cost centers (orders and WBS elements) selected in Allevo-Multi are displayed in columns and the cost elements are displayed row by row.

In this case, there is an aggregation of read and planning columns of different table sheets, whereby a selection list can be used to control which data (year, version, etc.) of the cost centers are to be displayed next to each other (see Figure 12.2).

|    |        |                      | 1000              | 1110           | 1200           |        |
|----|--------|----------------------|-------------------|----------------|----------------|--------|
|    |        |                      | Pfähler           | Kuhn           | Hertwig        |        |
|    |        |                      | Corporate Service | Vorstand       | Kantine        |        |
|    |        |                      | IPP_0000001000    | IPP_0000001110 | IPP_0000001200 |        |
| 12 | 420000 | Direct labor costs   | 1.094.336         | 57.524         | 999.999        | 36.813 |
| 15 | #      | Wages                | 1.094.336         | 57.524         | 999.999        | 36.813 |
| 16 | 430000 | Salaries             | 1.080.692         | 168.633        | 912.058        |        |
| 19 | #      | Salaries             | 1.080.692         | 168.633        | 912.058        |        |
| 24 | 435000 | Annual Bonus         | 101.657           | 17.875         | 81.226         | 2.556  |
| 27 | 440000 | Legal social expense | 220.162           | 49.019         | 161.364        | 9.779  |
| 32 | 449000 | Other pers. costs    | 11.200            | 638            | 10.000         | 562    |
| 33 | #      | Other Personal Costs | 333.020           | 67.532         | 252.590        | 12.898 |
| 34 | ##     | Personnel Costs      | 2.508.048         | 293.690        | 2.164.647      | 49.711 |
| 35 | 400000 | Raw Materials 1      | 170               | 170            |                |        |
| 42 | 403000 | Operating Supplies   | 51.405            | 20             |                | 51.385 |
| 50 | #      | Material Costs       | 51.575            | 190            |                | 51.385 |
| 52 | 481000 | Cost-acctg deprec.   | 556.495           | 550.332        | 2.180          | 3.983  |
| 56 | 483000 | Imputed interest     | 1.089.408         | 1.085.217      | 1.079          | 3.112  |
| 59 | #      | Imputed Costs        | 1.645.903         | 1.635.549      | 3.260          | 7.095  |
| 65 | 416300 | Water                | 12.332            | 12.332         |                |        |
| 72 | 451000 | Building maintenance | 25.692            | 25.692         |                |        |
| 77 | #      | External Services    | 38.025            | 38.025         |                |        |

Figure 12.2: Main view of totals sheet with data

The totals table is structured in a way that it can be set up with relative ease. Usually it is enough to simply adjust the number of rows and the formatting of the |ALLEVO| sheet.

**Note:** The macro based totals sheet is part of the Standard Allevo Master and cannot be manually created like the formula based totals sheet.

The assignment of the displayed columns in the total sheet is done by naming the read columns or their absolute column number (OptionalPosition) in Excel. In addition, you should assign a name for the respective entry in the dropdown field under Select Description.

The totals sheet and the summary sheet

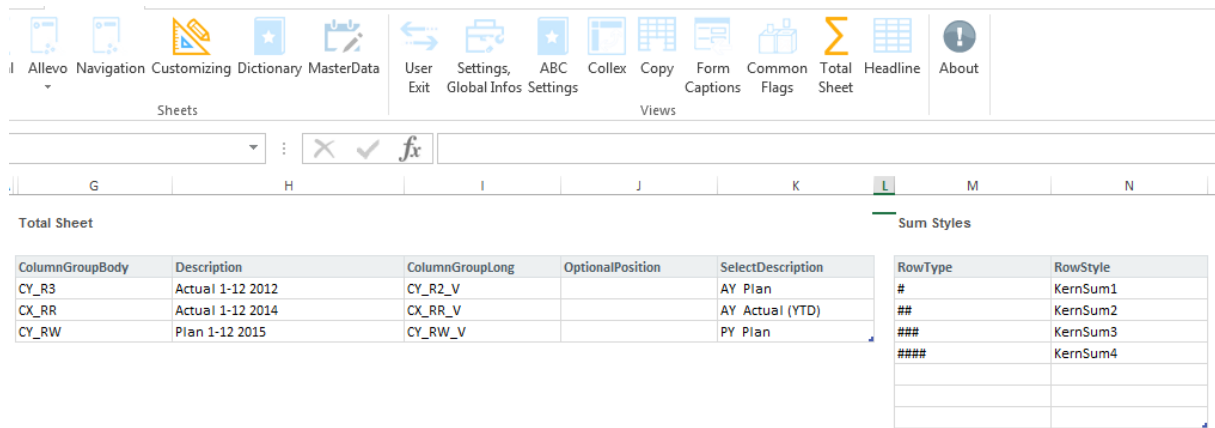


Figure 12.3: Totals sheet configuration

The assignments made in the configuration make it possible to leave the total sheet in the template free of any row or column structure, so that the user can concentrate on configuring the navigation and does not have to maintain the row structure twice. The row structure is copied from the name range CY\_KEYTOTAL from the Allevo sheet.

The grayed-out entries OBJECT | RESPONSIBLE | DESCRIPTION can be replaced by any local parameters (see info sheet) to display further information from the objects.

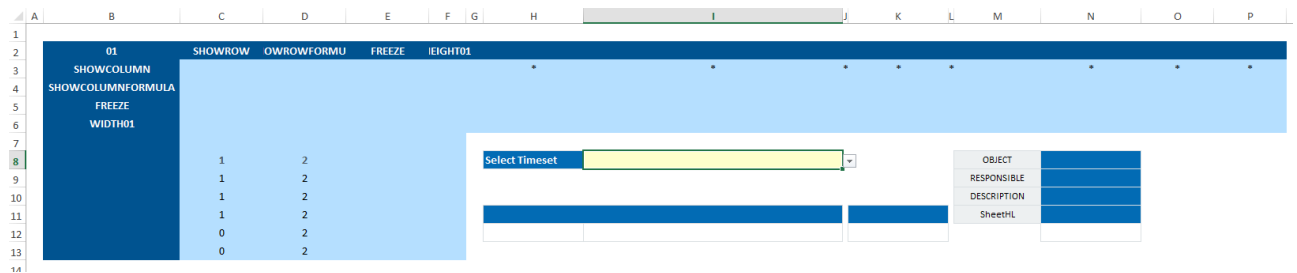


Figure 12.4: Empty totals sheet template

### 13 Allevo Tree: Object selection on the Excel side (only in ABC)

#### 13.1 Tree view for object selection in Excel

Before you can enter planning data, you must select the required object, depending on the object type. This selection is normally made via the called SAP transaction (using Inplace processing) or, in the case of the Allevo Business Client, via the panel.

If several objects are selected at this point (for example, by object group), it is possible to transfer this information from SAP to Excel in the form of a list, where it can be displayed in a Tree structure. You can then navigate further in Excel using the Tree view.

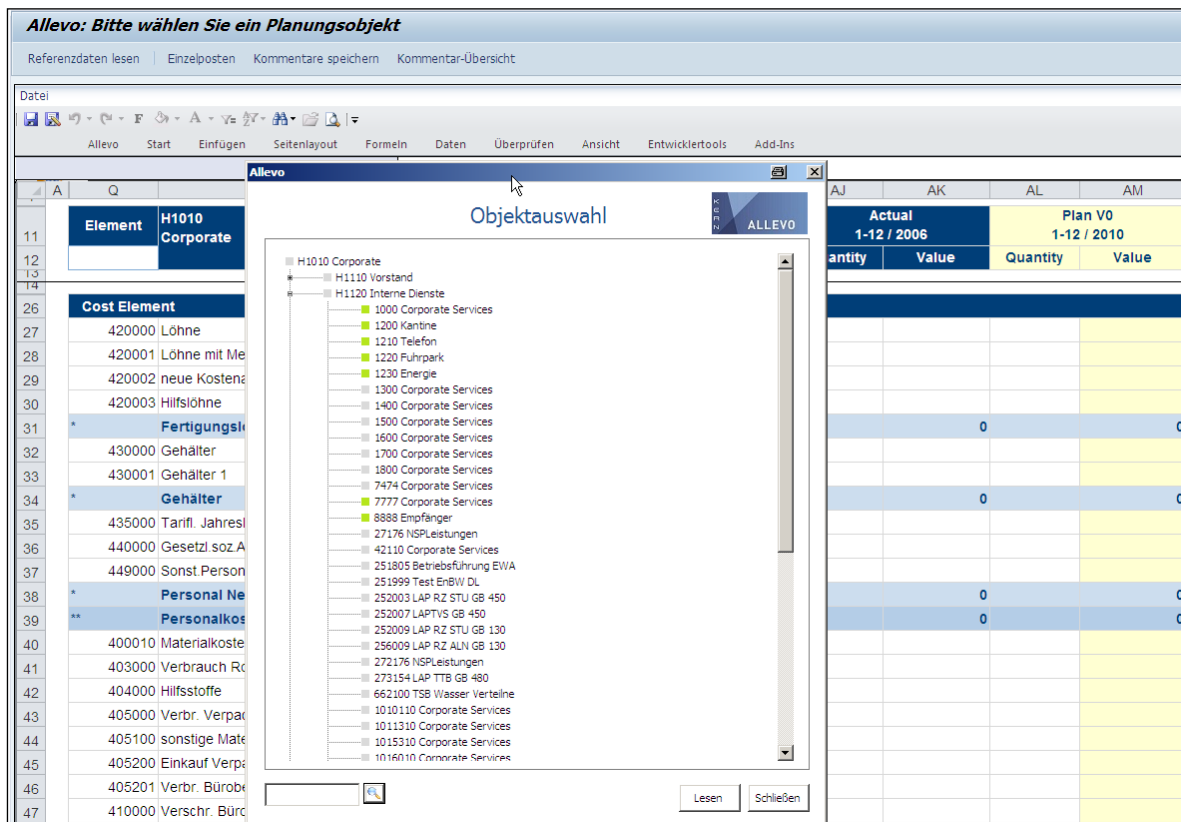


Figure 13.1: Object selection on the Excel side via Tree

The cost centers in the example in Figure 13.1 are marked in green if they can be planned based on the status and authorization of the user. Allevo automatically switches to reporting mode for cost centers marked in gray: in this case, planning data can no longer be transferred to SAP.

**Note:** Figure 13.1 shows the PopUp as it appears when using Inplace mode. When calling via Allevo Business Client (ABC), the Tree is part of the panel and has a slightly different look and functionality.

The search in the Tree PopUp is carried out using the short names of the objects and their descriptions. The placeholders '\*' and '?' are allowed in the search term. The search starts at the current node and automatically selects the next node found (use enter key or magnifying glass to continue searching).



Only after selecting an object via the [Read] button, the corresponding reference data are read from SAP and entered into the Allevo Master.

|              |   |
|--------------|---|
| <b>Note:</b> | For Allevo 3.1 and higher there is an additional information area available in the Tree, where individual customer notes can be stored. |
|--------------|---|

If the Tree is called for the objects of a group, which in turn has subordinate nodes in the hierarchy, the data for these groups can also be displayed (for example, to check the results of planning at the relevant group level). Optionally, summation can also be carried out using data in satellite tables (see constant GRP\_READ\_SATxx, also applies to MO transactions).

A green marking (= planning possible) appears for the group if a plannable object is defined in "1:n Customizing". In this case, however, the subordinate objects and nodes no longer appear in the Tree, because the data of these objects is already displayed in an aggregated form (this rule can be adjusted using the constant OBJ\_SEL\_IN\_EXCEL).

|              |   |
|--------------|---|
| <b>Note:</b> | As with classic processing in Allevo Single mode, only one object is copied from the Tree to the planning form at a time. If the Tree is activated in a layout with MO functionality, cross-object processing is also possible (see notes on MO in Chapter 15). |
|--------------|---|

## 13.2 Set up the Tree

The tree function is activated using the constant OBJ\_SEL\_IN\_EXCEL in the Allevo planning layout. In the Allevo Master, the ribbon icon in the navigation must also be set up; the corresponding macro is called SHOWTREEVIEW (see the F1 help for the constant).

|              |   |
|--------------|---|
| <b>Note:</b> | The Tree view uses the ActiveX-component MSCOMCTL.OCX, made available by Windows (e.g. in C:\Windows\SysWOW64 for Windows 7).<br><br>This component is not available in the 64-bit version of Excel. The Tree can therefore only be used with the 32-bit version. |
|--------------|---|

Text items in the Tree interface (buttons, headers) can be adapted to customer requirements (see Customizing sheet, table with heading "Forms and Controls", e.g. ControlName "IblTitle" for the heading of the Tree PopUp). To display the additional info area in the Tree, a text for the "IblInfo" control must be entered.

## 14 Allevo MultiObject

### 14.1 Functional overview

The term “MultiObject” first tells us that objects of different types can be read and planned with this function. MultiObject makes it possible to mix cost elements internal orders, WBS elements and profit centers within the same Excel planning file.

Since there is no grouping function available within SAP that work with multiple object types, starting with MultiObject is always controlled with a representative object. The available Allevo objects (KX) are especially useful for this purpose, but starting can also be done via a cost center or profit center as well.

Calling is done via the relevant MO transaction (e.g. transaction /ALLEVO/KXMO for Allevo objects or /ALLEVO/KSMO for cost centers).

In these cases, the representative object acts as a placeholder for a grouping of many objects of different types that cannot be stored in the SAP standard system: the representative object represents a list of objects and cost elements. How this list is prepared and saved is linked to form variants, which are described in the following chapters.

|              |   |
|--------------|---|
| <b>Note:</b> | Especially if the entry object does not have its own data as a placeholder, you can use the Allevo object for starting. |
|--------------|---|

Following variants can be distinguished when setting up a MultiObject Master:

(1) MultiObject with variable sheet assignment

For each object type, a separate Excel sheet is used, e.g. data for a cost center on the first sheet, and data for related projects on other sheets with a different structure.

This variant has advantages if many cost elements are combined with few objects; it is described in Chapter 17.

(2) Allevo MultiObject Mass (MOM)

Shows all posting combinations in the form of a list with all object types on the same Excel sheet (in the following referred to as “MOM”).

It is currently the most frequently used MO variant: the composition of the list is often taken from a satellite table, the content of which is provided, for example, by the additional module ProCED (see Chapter 15). The following two variants are special forms of the MOM list display.

(3) MOM with a call in MultiPage-mode (MOMP)

This variant allows you to create an MOM sheet for several representative objects at the same time, whereby the data for each representative object is stored on a separate Excel sheet. This is the ‘typical’ MultiPage mode, whereby each page can contain cross-object data (data from multiple object types): In this case, the call is made using Allevo's MultiPage transactions (such as transaction /KERN/ IPPKSM for cost centers). The necessary additional settings are described in Section 15.5.

(4) Allevo Multi-Object Dynamic (MOD)

MOD works in a special form of MOM: in this case, the list of posting elements is generated dynamically after calling the transaction in SAP: the user selects the required cost centers or cost elements, for example, and Allevo then generates a list of relevant combinations. This application is described in detail in Chapter 16. A possible way of generating this dynamic is to use the

ListGenerator ProCED (Provide Cost Element Dynamics), which is very valuable for WBS structures.

## 14.2 Authorizations

In all MultiObject cases you can only work with objects for which SAP authorization exists.

If the start object is not planned itself, you can skip the authorization check for it (see constant AUTH\_PLAN\_START).

## 14.3 Status management

Allevo status management applies to all planning objects involved. This is particularly important when changing the object type: Allevo searches for a planning layout with the same object ID code as when you access the object type to check the status. The layouts should therefore have been created, even if they are not to be directly planned.

An object that has not yet been released for planning or has already been completed cannot normally be planned using MultiObject (to handle differently, see constant STATUS\_READ\_ALL).

## 15 Allevo MultiObject Mass (MOM)

MultiObject Mass is probably the most commonly used form in MultiObject mode.

In general, MultiObject mode is characterized by the fact that different SAP object types can be mixed in one planning file, including objects such as orders, profit centers, cost centers, and WBS elements. The MOM also has the property that all objects can be edited in list form on only one worksheet.

### 15.1 General structure of the MOM planning template

In MOM, the Excel worksheet is no longer permanently linked to an SAP object. Instead, the object "slides" from the header data of the sheet into the planning line. The row layout is thus a combination of the planning object and the cost element to be planned (statistical key figure, activity type).

This general structure is already contained within the Allevo Standard Master (see Chapter 5 and Figure 15.1, columns F and G)

| Task List Object                             | Setclass | Element        | CC_OBJECT1         | CC_OBJECT2 | CC_OBJECT3 | CEC          | Actual 1-12 / 2009 | Actual 1-8 / 2010 | Plan V0 1-12 / 2010 |       |          |       |
|--|----------|----------------|--------------------|------------|------------|--------------|--------------------|-------------------|---------------------|-------|----------|-------|
|  |          |                |                    |            |            |              | Quantity           | Value             | Quantity            | Value | Quantity | Value |
| <b>Statistical Key Figure / Cost Element</b> |          |                |                    |            |            |              |                    |                   |                     |       |          |       |
| 1000   | 0101     | Cost Center    | 2010 Square Meters |            |            | Key Figure   |                    |                   |                     |       |          |       |
| 1000   | 0101     | Cost Center    | 420000 Wages       |            |            | Cost element |                    |                   |                     |       |          |       |
| 6000100                                      | 0103     | Internal Order | 420000 Wages       |            |            | Cost element |                    |                   |                     |       |          |       |
| 6000100                                      | 0103     | Internal Order | 420001 Overtime    |            |            | Cost element |                    |                   |                     |       |          |       |
| 6000100                                      | 0103     | Internal Order | 420002 Benefits    |            |            | Cost element |                    |                   |                     |       |          |       |
| T20301.1                                     | 0110     | WBS-Element    | 420000 Wages       |            |            | Cost element |                    |                   |                     |       |          |       |

Figure 15.1: General structure of the object/cost element list

**Note:** The systematics of the column structure remain unchanged with regard to the previous descriptions of Allevo in Single or MultiPage-mode.

It is characteristic of the MOM structure that different objects and object types are mixed on one sheet. In this case, it can be helpful to have additional information from the master data of the respective objects. Constant READ\_ELEMENT\_DATA can be helpful in this case.

### 15.2 Activating MOM in the settings

To use MOM, the following settings should be made under |Settings| of the worksheet |CUSTOMIZING| (according to default settings).

| Settings for Multi and MultiObject |        |  |        |
|------------------------------------|--------|--|--------|
| MOMDirect                          | WAHR   | MOM without assigned objects                         | WAHR   |
| NamingRule                         |        | Naming rule for Multi Page                           |        |
| CopyMultiSheet                     | FALSCH | Protect digital signature when creating multi sheets | FALSCH |
| UseMOMObjectType                   | WAHR   | Use object type                                      | WAHR   |

Figure 15.2: Settings for MOM

The MOM can also be used in MultiPage-mode; special settings are not necessary for this. Allevo recognizes this mode automatically from the called SAP transaction.

### 15.3 Form variants

The biggest challenge when using the MOM lies in the design of the form, i.e. how the list structure is generated. In principle, the following procedures apply here:

- Static structure directly via the Excel worksheet
- Dynamic structure via the use of a satellite, whereby the data stored there can be dynamically generated via additional Allevo functions (see notes on MOD and ProCED below).

#### 15.3.1 MOM form with static structure

In the static structure, the list of objects and cost elements is maintained directly in the |Allevo| worksheet in the desired structure. This variant has the advantage that it is quite easy to work with totals lines or links to satellites.

|              |   |
|--------------|---|
| <b>Note:</b> | In this case, it does not matter which object is used as a representative object in planning, since the same form is always called. The only alternative here is to control access to different Allevo Masters using Allevo planning layouts. |
|--------------|---|

One possible application for the static structure is, for example, the planning of maintenance projects that always remain the same in their structure and for which a specific planning layout can therefore be maintained. Another application could be a "fine tuning" of decentralized planning with regard to top-down specifications. The planning form could contain only the cost centers and orders to be planned, for example. The respective cost element is later selected, read and planned by a selection menu in Excel. Cost element specifications can be adjusted to all relevant planning objects in one go by adapting the preliminary planning.

The design of the MOM with fixed assignment is similar to that of the simple cost center Master: Cost elements and line definitions are entered directly. The planning object and its "Setclass / Object Type" (columns N and O) are also required.

#### 15.3.2 Dynamic structure via satellite

When working with a satellite, the desired form structure is not stored statically in the Excel planning file (Allevo Master), but in an SAP table (Allevo satellite). From there, the list is entered directly into the |MOM\_MASSEN| sheet when the Allevo Master is opened in SAP, depending of course on the representative object.

Access to an SAP table does is not really a dynamic option, but now it is possible to generate the list using additional programs and store it in the satellite table. In principle, any satellite can be used for this purpose. Satellite 0, however, is predestined for this purpose, as it only works read-only (and reading is always done first).

|              |   |
|--------------|---|
| <b>Note:</b> | Satellite 0 can be activated via constant SAT00_ACTIVE  |
|              | When using another satellite, you should be sure that the constant READ_ORDER_SAT is configured accordingly. The satellites must be read into the form before the cost element data is read, since this process provides the cost elements and objects to be read |

The corresponding satellite range must also be placed in the Allevo Master directly above the structure of the planning form (see Figure 15.3).

| Satellite14    |       |            |            |        |     |            |        |     |  |   |        |    |    |
|----------------|-------|------------|------------|--------|-----|------------|--------|-----|--|---|--------|----|----|
| Ins ROW        | KEY   |            |            |        |     |            |        |     |  |   |        |    |    |
| WRITE          | 7     |            |            | 8      |     |            |        |     |  | 9 | 11     | 12 | 13 |
| READ           | 7     |            |            | 8      |     |            |        |     |  | 9 | 11     | 12 | 13 |
| FORMULA        |       |            |            |        |     |            |        |     |  |   |        |    |    |
| FORMULATOVALUE |       |            |            |        |     |            |        |     |  |   |        |    |    |
| Dialogue01     |       |            |            |        |     |            |        |     |  |   |        |    |    |
|                | 01    | MO_ACO_AOT | YW         | YR     | YR2 | MW         | MR     | MR2 |  |   | CY_KEY |    |    |
|                | YEAR  | MO_ACO_AOT | MO_YWMO_YR | MO_YR2 |     |            |        |     |  |   | CY_KEY |    |    |
|                | YEAR  | MO_ACO_AOT | MO_YWMO_YR | MO_YR2 |     |            |        |     |  |   | CY_KEY |    |    |
|                | MONTH | MO_ACO_AOT | TYP        |        |     | MO_MMWO_MR | MO_MR2 |     |  |   | CM_KEY |    |    |
|                | MONTH | MO_ACO_AOT | TYP        |        |     | MO_MMWO_MR | MO_MR2 |     |  |   | CM_KEY |    |    |
|                | MONTH | MO_ACO_AOT | TYP        |        |     | MO_MMWO_MR | MO_MR2 |     |  |   | CM_KEY |    |    |
|                |       |            |            |        |     |            |        |     |  |   |        |    |    |
|                |       |            |            |        |     |            |        |     |  |   |        |    |    |
|                |       |            |            |        |     |            |        |     |  |   |        |    |    |
|                | X     |            |            |        |     |            |        |     |  |   |        |    |    |
|                |       | 1000       | KS         |        |     |            |        |     |  |   |        |    |    |
|                |       |            | KS         |        |     |            |        |     |  |   |        |    |    |
|                |       |            | KS         |        |     |            |        |     |  |   |        |    |    |
|                | 1     |            |            |        |     |            |        |     |  |   |        |    |    |
|                | 2     |            |            |        |     |            |        |     |  |   |        |    |    |

| 1000 |              | Kantine            |  | EC |
|------|--------------|--------------------|--|----|
|      |              | Hertwig            |  |    |
| #    | Cost Element |                    |  |    |
| 0    | 420000       | Direct labor costs |  | 1  |
| 0    | 420001       | Direct labor costs |  | 1  |

Figure 15.3: Satellite (here satellite 14) as a Standard MOD Example

In this example, Satellite14 is set to InsertNewExcelRows mode so that the complete Excel rows are copied for all satellite rows delivered by SAP. This automation allows you to plan any number of rows without having to prepare the Master for a large range.

Now we need to know how the desired list of objects and cost elements to be planned are brought into the satellite table and then used for the structure. Examples of possible procedures:

- Manual entry with the help of Allevo Shuttle
- The Allevo functions MOD and ProCED can generate very flexible structures and then introduce them into the satellite table. The MOD is explained in Section 15.7; ProCED is described in a separate manual.
- Optionally, customer specific programs may be used.

**Note:** Often, the necessary row definitions with the planning functions for year and month are determined via formulas, e.g. depending on the cost element type. In this case, a macro originally designed for MOD functions (Macro “startRowDef” see Chapter 16) may be helpful. Another macro, “startSum”, can create complete Excel totals formulas over any number of hierarchies and can also apply Excel cell format templates to totals rows.

### 15.4 MOM and the use of satellites

In many Allevo applications, satellite data is stored exactly for a single object. Since the MOM contains several objects on a single worksheet, it is only possible to read and write satellite data for each MOM planning object with certain restrictions.

However, a satellite range can also be explicitly assigned to the start object by appropriate placement: either on the table sheet |Allevo|, or, if available, on the table sheet that is assigned to the initial object (for example, |KST\_COSTCENTER|).

In special cases, however, constant GRP\_READ\_SATxx or exact row reading and writing may be useful.

### 15.5 MOM in MultiPage-mode (MOMP)

You can also call MOM with its list display for several representative objects simultaneously, whereby the data for each representative object is displayed on a separate Excel sheet. This is the typical MultiPage mode, in which each sheet can contain cross-object data: In this case, the call is consequently made using Allevo's MultiPage transactions (for example, transaction /KERN/ IPPKSM for cost centers).

Allevo automatically recognizes the MOMP-mode from the called SAP transaction and adjusts its functions accordingly.

If the additional module ProCED is also integrated in MultiPage mode, the constant MULTI\_PROCED must also be set, or the constant MULTIPAGE\_MODE must be set within ProCED (see notes on this in the F1 help).

The constant PRE\_SELECT should not be set if you are working on a single sheet with multiple object types: the preselection cannot check all possible combinations (for example, if only comments are entered for some cost elements).

If MOD functions are active at the same time, further restrictions apply (see Section 16.9).

### 15.6 MultiObject for PC planning over multiple company codes

The starting point is an Allevo Master with fixed sheets, in which each sheet reflects the data of a single company code.

The start object is the same for all company codes and may only be representative: different objects and object types can also be defined for each sheet.

The function is activated in the settings via the parameter FISH (Fixed SHEets).

If you are interested in this Master structure, please contact our support team.

### 15.7 Optional: MOM with an explicit entry range (One Page Multi)

One advantage of the list display in MOM is the performance with which data is transferred from SAP to Excel. This is especially advantageous if many objects are selected at the same time, for which a separate sheet would have to be created in MultiPage mode.

On the other hand, editing per object is often more intuitive and easier for the user to understand. In order to combine both of these advantages, Allevo offers the option of processing data from the list using an explicit input area for each object: in this case, the user chooses a planning object (such as a cost center) from the list of all objects transferred from SAP to Excel. An Excel macro copies the corresponding planning data from the MOM data range to a fixed input area. Modified or newly entered values are finally written back to the MOM data range.

When executing the function "transfer plan data", all values are transferred from the MOD-data range to SAP (for all projects simultaneously).

**Note:** This function is currently not contained in the Allevo Standard Master. If you are interested, please contact our support team.

The basic structure of the input range corresponds to an Allevo Standard Master with a fixed cost element structure. Here is an example for a cost center:

| H1000            | OnePageMulti | Nullzeilen aus             | CE TYPE | BU YTD 2016/2017 | AC YTD 2016/2017 | PY Full 2016/2017 | BU Full 2016/2017 | F1 Full 2016/2017 | AC Feb     | AC Mar     |
|------------------|--------------|----------------------------|---------|------------------|------------------|-------------------|-------------------|-------------------|------------|------------|
|                  | H1000        | IDES Deutschland – BK 1000 |         |                  |                  |                   |                   |                   |            |            |
| <b>Kostenart</b> |              |                            |         |                  |                  |                   |                   |                   |            |            |
| #                | OAS_WAGES    |                            |         | 1.556.961        | 0                | 3.251.554         | 3.251.554         | 0                 | 207.609    | 195.238    |
| #                | OAS_SALAR    |                            |         | 2.100.440        | 0                | 4.362.754         | 4.362.754         | 0                 | 271.750    | 286.208    |
| #                | OAS_P_IMP    |                            |         | 882.122          | 0                | 2.078.116         | 2.078.116         | 0                 | 125.861    | 116.917    |
| ##               | OAS_PERS     |                            |         | 4.539.523        | 0                | 9.690.423         | 9.690.423         | 0                 | 605.220    | 598.361    |
| #                | OAS_MAT      |                            |         | 448.374          | 0                | 1.102.985         | 1.102.985         | 0                 | 52.965     | 52.745     |
| #                | OAS_IMP      |                            |         | 1.200.649        | 3.268.379        | 2.069.440         | 2.069.440         | 0                 | 171.164    | 171.104    |
| #                | OAS_EXT      |                            |         | 634.319          | 0                | 1.604.617         | 1.604.617         | 0                 | 73.377     | 73.377     |
| #                | OAS_TAX      |                            |         | 4.368            | 0                | 26.206            | 26.206            | 0                 | 0          | 0          |
| #                | OAS_OTHERS   |                            |         | 1.024.221        | 0                | 1.926.653         | 1.926.653         | 0                 | 140.680    | 140.514    |
| ##               | OAS_OTHER    |                            |         | 3.311.932        | 3.268.379        | 6.729.901         | 6.729.901         | 0                 | 438.186    | 437.740    |
| ###              | OAS_PPRM     |                            |         | 7.851.455        | 3.268.379        | 16.420.324        | 16.420.324        | 0                 | 1.043.406  | 1.036.100  |
| #                | OAS_JCA      |                            |         | -9.767.220       | 0                | -16.668.974       | -16.668.974       | 0                 | -1.943.473 | -1.404.245 |
| #                | OAS_ORDERS   |                            |         | 166.932          | 0                | 297.101           | 297.101           | 0                 | 23.711     | 23.965     |
| ##               | OAS_ALLOC    |                            |         | -9.600.287       | 0                | -16.371.873       | -16.371.873       | 0                 | -1.319.761 | -1.390.280 |
| ###              | OAS_SEC      |                            |         | -9.600.287       | 0                | -16.371.873       | -16.371.873       | 0                 | -1.319.761 | -1.390.280 |
| ####             | OAS          |                            |         | 7.851.455        | 3.268.379        | 16.420.324        | 16.420.324        | 0                 | 1.043.406  | 1.036.100  |

Figure 15.4: MOD with the display of a cost element hierarchy

Via the *Select Cost Center* button, a cost center or node is called. While it is being called, values are written in the *data range* and the values of the called cost centers are written in the *planning interface*.

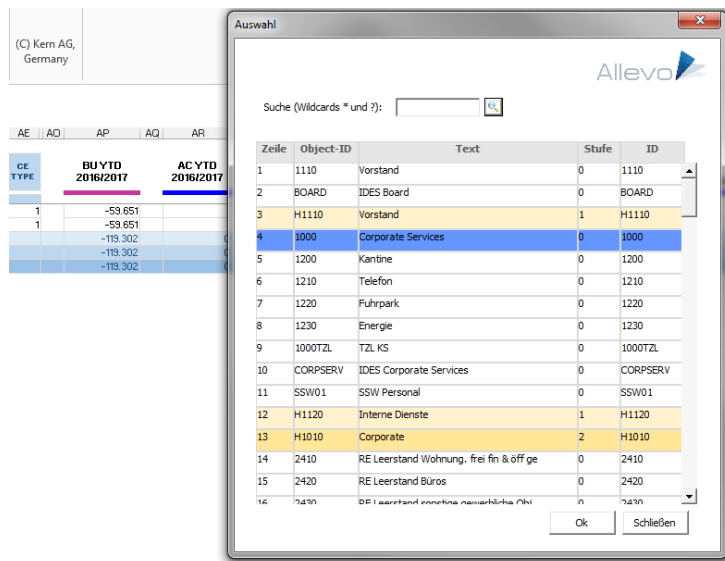


Figure 15.5: MOD with the selection of cost centers

The cost element hierarchy is permanently stored in the input area. Therefore, the MOM data area does not require any totals rows and therefore no totals formulas (that are usually used in the MOD variant; can be switched off using constant DYN\_SAT\_CHAR if necessary).

Usually a satellite is stored in the data area, which is dynamically provided with the relevant posting rows via MOD functions or ProCED (see Section 15.3.2).



## 16 Allevo MultiObject Dynamic (MOD)

### 16.1 Functional overview

MOD is a special form of MOM with the aim of being able to select objects and cost elements from Allevo's start screen. The planning form then automatically generates a structure from the combination of objects and cost elements, taking into account rows for subtotals at the appropriate places.

|              |   |
|--------------|---|
| <b>Note:</b> | In current Allevo versions, MOD also supports statistical key figures. For the sake of simplicity, in the following section we will be dealing with the use case for cost elements. In most cases the descriptions provided here also apply to statistical key figures. |
|              | MOD also supports profit centers as an object type. In this case, accounts can be used in the same way as cost elements.  |

There are two basic variants available:

- Structure based on object hierarchy (using cost center or order)
- Structure based on cost element hierarchy

Corresponding to these variants, the appropriate sort levels and subtotals are generated automatically. The following section explains both procedures using examples.

|              |   |
|--------------|---|
| <b>Note:</b> | MOD supports all Allevo object types except PR (= projects / WBS elements). It is activated via corresponding constants on the SAP side and suitable macros in the Allevo Master (see constant DYN_KSTAR_SAT) |
|              | The advantages of MOD and ProCED can also be combined, for example, to obtain additional information on the posting relationships (see notes in Section 16.8).  |

### 16.2 Examples with the MOD planning input screen

#### 16.2.1 Sorting by object

Sorting by object uses a similar structure to the MOM Master and corresponds to the following schematic:

```

4711 Kostenstelle
1000 Kostenart 1
2000 Kostenart 2
*Summe
4712 Kostenstelle 2
1000 Kostenart 1
2000 Kostenart 2
*Summe
**Gesamtsumme

```

Figure 16.1: MOD with symbolic list of cost elements and totals

In the following example, a cost center group (with two cost centers) and a cost element group are entered in the Allevo start screen. The cost element group has two subgroups, which each contain two cost elements. When sorting by cost element, the following display appears in Allevo:

| Object | 2100   | Finanzen & Admin.<br>Nagy      | Plan 0 1-12 / 2010 |                | Plan 0 1-12 / 2013 |                  | Plan 0 1-12 / 2013 |          |
|--------|--------|--------------------------------|--------------------|----------------|--------------------|------------------|--------------------|----------|
|        |        |                                | Quantity           | Value          | Quantity           | Value            | Quantity           | Value    |
|        |        | <b>Cost Element</b>            |                    |                |                    |                  |                    |          |
| 2100   | 420000 | Fertigungs-Loehne              |                    | 30.000,0       |                    | 113.160,0        |                    |          |
| 2100   | 421000 | Hilfs-Loehne                   |                    |                |                    | 10.000,0         |                    |          |
| 2100   | *      | <b>G2A Löhne</b>               |                    | <b>30.000</b>  |                    | <b>123.160</b>   |                    | <b>0</b> |
| 2100   | 430000 | Gehaelter                      |                    | 859.831,7      |                    | 859.831,7        |                    |          |
| 2100   | 435000 | Tariff. Jahresleist.           |                    | 61.783,0       |                    | 61.783,0         |                    |          |
| 2100   | *      | <b>G2B Gehälter</b>            |                    | <b>921.615</b> |                    | <b>921.615</b>   |                    | <b>0</b> |
| 2100   | **     | <b>G1 Summe Löhne Gehälter</b> |                    | <b>951.615</b> |                    | <b>1.044.775</b> |                    | <b>0</b> |
| 2101   | 420000 | Fertigungs-Loehne              |                    |                |                    | 5.000,0          |                    |          |
| 2101   | 421000 | Hilfs-Loehne                   |                    |                |                    | 17.000,0         |                    |          |
| 2101   | *      | <b>G2A Löhne</b>               |                    | <b>0</b>       |                    | <b>22.000</b>    |                    | <b>0</b> |
| 2101   | 430000 | Gehaelter                      |                    |                |                    | 10.100,0         |                    |          |
| 2101   | 435000 | Tariff. Jahresleist.           |                    |                |                    | 7.000,0          |                    |          |
| 2101   | *      | <b>G2B Gehälter</b>            |                    | <b>0</b>       |                    | <b>17.100</b>    |                    | <b>0</b> |
| 2101   | **     | <b>G1 Summe Löhne Gehälter</b> |                    | <b>0</b>       |                    | <b>39.100</b>    |                    | <b>0</b> |

Figure 16.2: Example of MOD output after sorting by object

All cost elements of the selection are assigned to each cost center; totals rows are automatically inserted for the subgroups. A grand total is also displayed for each cost center.

### 16.2.2 Sorting by cost elements

This type of sorting is recommended if single cost elements are to be planned for multiple objects. Again, here is the basic structure:

|                         |
|-------------------------|
| <b>1000 Kostenart 1</b> |
| 4711 Kostenstelle 1     |
| 4712 Kostenstelle 2     |
| <b>*Summe</b>           |
| <b>2000 Kostenart 2</b> |
| 4711 Kostenstelle 1     |
| 4712 Kostenstelle 2     |
| <b>*Summe</b>           |
| <b>**Gesamtsumme</b>    |

Figure 16.3: MOD with cost elements and assigned cost centers

In the Allevo planning screen, the following is displayed (example):

| Object | 2100   | Finanzen & Admin.<br>Nagy     | Plan 0 1-12 / 2010 |               | Plan 0 1-12 / 2013 |                | Plan 0 1-12 / 2013 |          |
|--------|--------|-------------------------------|--------------------|---------------|--------------------|----------------|--------------------|----------|
|        |        |                               | Quantity           | Value         | Quantity           | Value          | Quantity           | Value    |
|        |        | <b>Cost Element</b>           |                    |               |                    |                |                    |          |
| 2100   | 420000 | Fertigungs-Loehne             |                    | 30.000,0      |                    | 113.160,0      |                    |          |
| 2101   | 420000 | Fertigungs-Loehne             |                    |               |                    | 5.000,0        |                    |          |
|        | *      | <b>GRP1 Fertigungs-Loehne</b> |                    | <b>30.000</b> |                    | <b>118.160</b> |                    | <b>0</b> |
| 2200   | 420000 | Fertigungs-Loehne             |                    | 9.863,9       |                    | 34.863,9       |                    |          |
| 2300   | 420000 | Fertigungs-Loehne             |                    | 10.738,6      |                    | 25.738,6       |                    |          |
|        | *      | <b>GRP2 Fertigungs-Loehne</b> |                    | <b>20.603</b> |                    | <b>60.603</b>  |                    | <b>0</b> |
|        | **     | <b>TOT Fertigungs-Loehne</b>  |                    | <b>50.603</b> |                    | <b>178.763</b> |                    | <b>0</b> |
| 2100   | 421000 | Hilfs-Loehne                  |                    |               |                    | 10.000,0       |                    |          |
| 2101   | 421000 | Hilfs-Loehne                  |                    |               |                    | 17.000,0       |                    |          |
|        | *      | <b>GRP1 Hilfs-Loehne</b>      |                    | <b>0</b>      |                    | <b>27.000</b>  |                    | <b>0</b> |
| 2200   | 421000 | Hilfs-Loehne                  |                    |               |                    | 25.000,0       |                    |          |
| 2300   | 421000 | Hilfs-Loehne                  |                    |               |                    | 10.000,0       |                    |          |
|        | *      | <b>GRP2 Hilfs-Loehne</b>      |                    | <b>0</b>      |                    | <b>35.000</b>  |                    | <b>0</b> |
|        | **     | <b>TOT Hilfs-Loehne</b>       |                    | <b>0</b>      |                    | <b>62.000</b>  |                    | <b>0</b> |

Figure 16.4: Example MOD output when sorting by cost element

The rows with the subtotals (GRP1, GRP2) correspond to the subgroups of cost centers: for every two cost centers in this example, groups are summarized on the SAP side.

### 16.2.3 Additional sorting options

The two examples above correspond to an object/cost element structure according to methods (1) and (2) in the constant DYN\_KSTAR\_SAT. Further options (including combined totals) are described in the F1 help for this constant.

### 16.2.4 Display in BAB-view

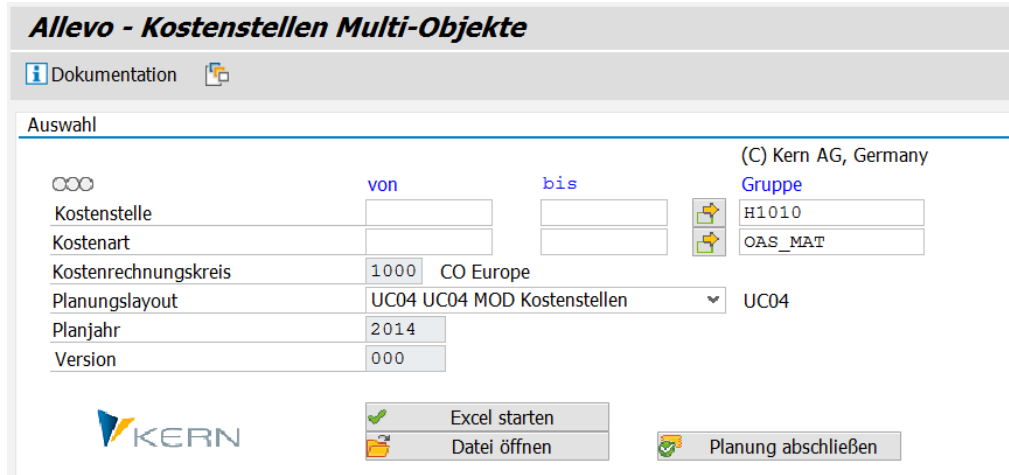
With additional functions in the Allevo Master, it is possible to display the data transferred from SAP in a so-called BAB-view: i.e. a list of cost elements/accounts with relevant objects or object groups in the columns next to each other (similar to the totals sheet in MultiPage mode).

This view can be particularly useful in Allevo reporting; it is described in detail in Section 16.12 below.

## 16.3 Functioning principle for cost elements

### 16.3.1 Calling the selection input screen

Allevo-MOD is called using the usual MultiObject transactions (for example, /ALLEVO/KSMO when starting Allevo-MOD via cost center). If MOD constants are active, the following start screen appears:



**Allevo - Kostenstellen Multi-Objekte**

Dokumentation

Auswahl

(C) Kern AG, Germany

von bis Gruppe

Kostenstelle    H1010

Kostenart    OAS\_MAT

Kostenrechnungskreis 1000 CO Europe

Planungslayout UC04 UC04 MOD Kostenstellen UC04

Planjahr 2014

Version 000

Excel starten

Datei öffnen

Planung abschließen

Figure 16.5: MOD start screen

Multiple objects and cost elements as well as corresponding groups can be entered in the Allevo start screen.

A cost element group is dissolved as usual into the corresponding cost elements and transferred according to the defined hierarchy. One exception is aggregated reading of groups via Allevo's 1:n assignments (see next Section)

**Note:** Optionally, you can define the required cost elements as fixed. The constant DYN\_KSTAR\_FIX is provided for this purpose. In this case, the input fields for the cost element are not displayed on the selection screen.

When calling the planning transaction, Allevo first checks how the list of posting combinations is to be sorted: a rough distinction is made between the options for sorting by object or cost element hierarchy.

**Note:** The MOD is activated with the constant DYN\_KSTAR\_SAT for the respective planning layout. The "Value from" column should contain the number of the satellite, which transfers the structure of the posing combinations (from objects and cost elements).

The number in the "Value to" field defines how sorting should be carried out: e.g. via objects (1) or via cost elements (2). Further options are described in the F1 help for the constant.

In current Allevo versions, it is possible to display two different sorting methods in the same Allevo Master (on separate sheets, see the special function described below or the F1 help for the constant).

In the second step, Allevo determines all posting combinations from object and cost elements according to the desired sorting option. Special functions:

- You can restrict this list to only include combinations for which postings are already stored in SAP for the reference period in question (see constant DYN\_KSTAR\_PRE).
- When starting with an object group, it can be useful to restrict the list of objects created using other criteria (for example, restrict it to a certain level of the hierarchy or to consider 1:n groups). These functions must be set up using constant GRP\_LEVEL\_RESOLVE. See Section 16.5.3 for more details.

Via the MOD satellite, all posting combinations are transferred to the Allevo Master: the field structure for this MOD satellite is predefined (see settings below). The data in this satellite is only temporarily available for transmission to Excel; it is not permanently stored in the satellite table.

|              |  |
|--------------|--|
| <b>Note:</b> | The MOD-satellite is activated with the constant DYN_KSTAR_SAT; it does not require a license. We recommend using a separate Allevo planning layout for MOD. |
|--------------|--|

In addition to the specifications for the object and cost elements (including cost element type), the MOD satellite also provides specially selected rows in which totals information can be displayed. This information is further processed using two macros in the Excel Master. Related functions:

- Determination of the appropriate row definitions for yearly and monthly planning via cost element type or cost element
- Formatting totals rows
- Entering totals formulas based on the totals structure

### 16.3.2 Aggregation according to cost element group

In two variants, MOD can generate rows with aggregated data for each cost element group (whereby the ID code of the group is directly transferred to Excel):

- Starting via a cost element group, whereby the subordinate groups are entered in the Allevo 1:n settings. Before Allevo 3.4 this was only possible when sorting by cost element (i.e. an entry of 2, 4, or 6 in DYN\_KSTAR\_SAT).
- Entering groups for the constant DYN\_KSTAR\_FIX (option 3 or 4).

In this case, MOD treats the group as a single cost element: only one row is created in the Excel sheet with the name of the group (that is, without breaking down or dissolving the corresponding hierarchy). "99" is entered as the cost element category. When the reference data is read, the values of all associated cost elements are automatically aggregated (corresponds to the usual Allevo function).

|              |   |
|--------------|---|
| <b>Note:</b> | Normally, summation functions are also defined for the group on the Excel side (usually controlled via cost element type 99). The summary of the group in a single row (described above) may interrupt the summation functions on the Excel side (should be checked on a case-by-case basis). |
|--------------|---|

## 16.4 Functioning principle for statistical key figures

The planning input screen for statistical key figures is automatically filled here according to the same procedure as for cost elements, but without group totals rows. The data is transferred again via an independent satellite (see constant DYN\_STAGR\_SAT).

In contrast to the functioning principle for cost elements, statistical key figures cannot be selected in the Allevo start screen; instead, you can define a group with the relevant statistical key figures using the constant DYN\_STAGR\_GRP. If this constant is not maintained, Allevo transfers all statistical key figures defined in the SAP system.

|              |   |
|--------------|---|
| <b>Note:</b> | The functions for statistical key figures supplement the breakdown by cost element (can only be used simultaneously). |
|--------------|---|

## 16.5 Special functions

### 16.5.1 Calling MOD via the ABC Tree

MOD can also be used in combination with the ABC-Tree: object selection at the start determines the hierarchy that is displayed in the Tree. An object or group selected in the tree determine the functions in MOD.

When using MOD in this way, the list of the desired cost elements must be stored in a fixed manner (see constant DYN\_KSTAR\_FIX).

This function is currently unavailable for the Inplace Tree (Tree and MOD can only be used alternatively).

### 16.5.2 Additional lists with MOD base information

In MOD, relevant posting combinations are dynamically transferred to the Allevo Master as a list. If DYN\_KSTAR\_PRE is also active, this list only contains combinations that already have values in the SAP system. In this situation, it can be helpful to have a list of objects (for example, cost centers) or cost elements in Excel that were determined by the original selection on the SAP side (for example, as selection help when entering new posting combinations).

This complete list of objects or cost elements is created using the constant DYN\_SAT\_OBJECTS. The constant uses one or more other satellites whose structure is predefined (structure /KERN/IPP\_S\_DYN\_OBJECT\_SAT).

The optionally delivered lists concern objects, cost elements, and statistical key figures. Additional options:

- The list contains ID codes and descriptions in the login language
- Transfer of the level within the hierarchy
- Note as to whether the current entry contains postings or not
- Multiple lists can be transferred together in a predefined sequence

For a more detailed explanation of the constant, see the F1-documentation.

### 16.5.3 Breakdown of groups to obtain objects

In many cases, Allevo with MOD functions is called via an object group on from the start screen: Allevo breaks down this group and then displays the assigned objects (such as cost centers) in Excel with totals levels corresponding to the hierarchy within the initial group.

In some cases, it may be useful to adapt the display of the hierarchy to specific requirements. The constant GRP\_LEVEL\_RESOLVE is available with the following basic functions:

- The list of objects (such as cost centers) can be restricted to a certain level of the hierarchy. This can be useful for reporting purposes, for example: Allevo reads the values summed to the groups of a given level where necessary. The objects in a lower level are not listed in Excel.
- The constant also takes 1:n assignments into account: such a group is not broken down, but is treated as a planning object. The MOD therefore also supports Allevo planning at the group level (via representative objects).

These functions cannot be used for every sorting scenario. See the F1 help for the constant for more details.

#### 16.5.4 Use different sorting types / views simultaneously in Excel

In current Allevo versions, it is possible to display several different sorting types simultaneously in Excel. The desired views are transferred to Excel one after the other and inserted on different sheets. Of course, you can still only enter or plan on one of the sheets.

This function can serve as an enhancement of Allevo reporting. For details, see the F1 help for the constant DYN\_KSTAR\_SAT.

#### 16.5.5 Sheet comments

The recording of data in MOD does not differ from other Allevo processing functions. Only one sheet comment is saved for the object group (instead of for a representative object). This means that if you change the structure of the group, the comments are retained.

#### 16.5.6 Close planning

When the function |Close planning| is called, all objects from the start selection (e.g. an object group) that previously had a status of 3 are closed.

### 16.6 Relevant constants and settings in SAP

A satellite is used to transfer data from SAP to Excel. In contrast to the usual procedure, this satellite can also work without an append, since the data is only provided to Excel at the time of transmission and is not stored in the satellite table.

**Note:** Working without a satellite append is possible for Allevo versions 3.3 and higher: this holds true for both cost centers and statistical key figures. In this way, the setup of MOD functions is simplified.

If there is no append defined for the satellite table, Allevo uses the standard structure according to the structures /KERN/IPP\_S\_DYN\_KSTAR\_TAB for cost elements and /KERN/IPP\_S\_DYN\_STAGR\_TAB for statistical key figures. In particular, the 5 fields that are minimally required for MOD functions and which must be activated as per usual in the Excel Master are stored there.

In some use cases, an individual satellite structure may be useful. For example, if MOD is working together with ProCED and further information on posting combinations should be transferred from there. However, even in such cases, the first fields should still correspond to the minimally required structure.

|              |   |
|--------------|---|
| <b>Note:</b> | The minimally required structure is stored in /KERN/IPP_S_DYN_KSTAR_SAT and is meant as a copy template for individual appends.                       |
|              | Please do NOT include the structure directly in the append of the satellite table. This will help maintain compatibility with future Allevo versions. |

The satellite number must be entered for the constant DYN\_KSTAR\_SAT: this activates the MOD functions on the SAP side.

The functions of the MOD are largely controlled by constants. Here is an overview of relevant constants:

| Constant for cost elements | Meaning   |
|----------------------------|---|
| DYN_KSTAR_SAT              | This is the central constant for activating MOD: it contains the indication of the satellite used for transferring objects and cost elements and specifications for sorting the list on the Excel side. |
| DYN_KSTAR_SAT_SORT         | Define sequence for transferring to Excel (by object group)   |
| DYN_KSTAR_SAT_RANGE        | Control time range for reading old data   |
| DYN_KSTAR_SAT_STATUS       | Observe Allevo object status when transferring posting combinations   |
| DYN_KSTAR_FIX              | Work with predefined cost elements/accounts or groups   |
| DYN_KSTAR_VALUES           | Restrict the cost element selection via authorizations or defined groups.   |
| DYN_KSTAR_PRE.             | Reduces the cost element list to entries that already have postings saved in SAP; applies automatically for statistical key figures.  |
| DYN_KSTAR_ROWS             | The number of reserved rows in Excel can be controlled  |
| DYN_SAT_CHAR               | Special characters for displaying group levels. Totals rows can be completely deactivated if desired (e.g. when working with ProCED or displaying using OAAS)   |

| Constant Stat. key figures | Meaning  |
|----------------------------|--|
| DYN_STAGR_SAT              | Define the satellite used for transferring objects with statistical key figures. |
| DYN_STAGR_GRP              | Group with the statistical key figures that are to appear in the Allevo Master.  |
| DYN_STAGR_ROWS             | The number of rows reserved in Excel can be controlled                           |

The following constants may also be useful in some cases:

| Constants other   | Meaning   |
|-------------------|---|
| GRP_LEVEL_RESOLVE | Restrict breakdown of object hierarchies to a predefined level (for example, for reporting functions such as the BAP view).<br>As of Allevo 3.4, this function can also be used for cost element groups.<br>If necessary, READ_GROUPS_FROM_SAP must also be active. |
| MULTI_WITH_GROUPS | Read reference data on the object group level   |



|                   |   |
|-------------------|---|
| DYN_SAT_OBJECTS   | Transfer an additional list with basic information for the selected objects (e.g. cost center) and cost elements. |
| READ_STATUS       | Show the status of the respective object via an additional comment column   |
| READ_ELEMENT_DATA | Expand the master data of the planning object (same as MAP_FIELDxx, but in this case by row)                      |

## 16.7 Settings on the Excel side

### 16.7.1 Setting up the MOD satellite

In the controlling area of the Master, the ranges SatXXRow and SatXXRW need to be created in correspondence with the append structure as it exists in the satellite table on the SAP side (see figure below).

Figure 16.6: Satellite range (satellite 14 as an example)

Such ranges must exist individually for cost elements and statistical key figures, because two different satellites are also used. When the Allevo Master is called, the relevant objects and associated cost elements are automatically entered in the data range of the satellite. The range should be selected either as large as the max. number of rows, or one of the dynamic line extension modes, such as insertNewSatRows or insertNewExcelRows, should be configured (as above).

**Note:** Only relevant for the standard execution mode: If the provided number of rows is entered in the constant DYN\_KSTAR\_ROWS, Allevo issues a warning message at the start if the number of rows in the current selection is exceeded (or DYN\_STAGR\_ROWS for statistical key figures).

The structure of the satellite table and the arrangement of the columns for the display is fixed (as usual, columns 1 to 7 are reserved for index fields):

- (8) Number/ID code of the planning object (or the special character for totals rows)
- (9) Description of the planning object
- (10) Setclass when starting (e.g. 0101 when calling via cost centers)

- (11) ID code of the cost element / statistical key figure or the special character for totals rows)
- (12) Description of cost element or statistical key figure
- (13) Cost element type

The columns named above need to be shown in Excel in order to use the basic functions of MOD. The information in the following columns is optional, but may be useful in some cases:

- (14) is set with X if the posting object comes from old data.
- (15) is set with X if the row contains a top cost element group.
- (16) similar to (8), but for totals rows the ID code of the group is provided.
- (17) similar to (11), but for totals rows the ID code of the cost element group is provided.

The last two options are only available as of Allevo 3.4, and allow groups and posting elements to be mixed in a single column. In the standard columns (8) / (11), totals rows are designated with special characters (see details in next section).

### 16.7.2 Configuring StructureBuilder for MOD

The StructureBuilder defines the properties of the respective rows (e.g. row definition depending on cost element type, applied via macro "startRowDef"), as well as the desired totals and cell formatting (applied via the macro "startSum").

The corresponding settings are made on the |Structure| worksheet (see example on this sheet, which is largely self-explanatory).

### 16.8 MOD and ProCED

When working together, MOD and ProCED make the following functions possible:

- As described above, MOD uses a satellite table to transfer the relevant planning elements from SAP to Excel. Normally the field structure of this satellite is fixed (see Section 16.7). In combination with ProCED, this satellite can include additional information that is available via the ProCED field mapping (for example, additional information from the master record of the respective object). The fields that can be used in the Allevo Master are derived from the field mapping in ProCED.
- ProCED checks whether data records already exist in the SAP system for the intended planning elements (similar to constant DYN\_KSTAR\_PRE, which no longer applies in this case).
- Sorting and subtotals are handled in the same way as with MOD.

MOD automatically recognizes when ProCED is active (see constant ACTIVE\_PROCED or functions in the "Sat.Assistant"). This results in the following procedure:

- The selection characteristics that the user selected when entering MOD are transferred to ProCED (for example, a cost center group and selection of cost elements).
- ProCED uses this information to determine the list of relevant planning elements.
- The MOD adopts this list, sorts it according to the current constants and then creates the rows for subtotals.

The coupling of these two applications can be used for cost elements (constant DYN\_KSTAR\_SAT) and statistical key figures (DYN\_STAGR\_SAT). Please also note the following points:

- Data exchange with ProCED is usually carried out via a satellite table. A suitable append must be created for the satellite table, which also contains the MOD base fields (as described in the previous sections, the fields must be filled using ProCED).
- The selection of the ProCED data itself is currently still carried out without consideration of the Allevo status.
- ProCED usually uses a representative object for data transfer. Alternatively, the data read by ProCED can also be stored and transferred individually for each object: The ProCED constant MULTIPAGE\_MODE must then be set, as well as the Allevo constant GRP\_READ\_SATxx on the Allevo planning side.
- When working with ProCED, it can be useful to work completely without group totals; for this purpose, constant DYN\_SAT\_CHAR provides an option.
- The Allevo constant DYN\_SAT\_OBJECTS can be helpful for providing a list of the objects involved; if necessary, it can be combined with the constant DYN\_KSTAR\_SAT\_STATUS to offer the user only objects with a status for selection on the Excel page.

|              |  |
|--------------|--|
| <b>Note:</b> | <p>The interaction of MOD and ProCED is primarily designed to generate data for those object types that are specified when starting via Allevo (e.g. KS when starting with a cost center)</p> <p>However, ProCED also offers options for reading additional data from other object types (e.g. controlled by constant READ_ASSIGNED_OR); such an application is only possible to a limited extent and must be checked in individual cases.</p> |
|--------------|--|

## 16.9 MOD and MultiPage-mode

In some cases, it may be useful to call the MOD (with its list display) for several representative objects at the same time, whereby the data for each representative object is displayed on a separate Excel sheet.

Here we are dealing with the typical MultiPage-mode, but in this case, each sheet can contain data for multiple objects: In this case, the call is made via the Allevo MultiPage transactions (for example, transaction /KERN/IPPKSM for cost centers).

Allevo automatically recognizes if the call is made via MultiPage-mode: the list of the cost elements is generated on the template sheet "Allevo" and are then copied to the individual sheets. The MOD specific macros *startSum* and *startRowDef* should already be called in the event MOD\_TEMPLATE, instead of being first called by the events AFT\_LEAD\_SAT or AFT\_READ (as is otherwise common).

As a result, the combination of MOD and MultiPage-mode can only be executed with the following restrictions:

- Cost elements must be permanently stored (see constant DYN\_KSTAR\_FIX)
- The object number (column MO\_AO) must be derived from formulas from the header data for each page (CC\_OBJECT1).
- The list of posting elements (cost elements/statistical key figures) is the same on all sheets, since it is only transferred once. The constant DYN\_KSTAR\_PRE should not be active.
- Further objects cannot be added via Excel functions

See also the notes on the offline mode in the next section.

## 16.10 MOD and offline process

Allevo supports the entry of plan data in offline mode. In this procedure, an Allevo planning transaction is automatically called several times, and the associated planning files are saved or read in with reference data (details are described in the Allevo & SAP manual).

Usually, offline-relevant objects are specified using multiple selection or an object group; that is, the same selection characteristics as when you call a single transaction with MOD in the background. Allevo must therefore decide how to display the corresponding objects in the Master.

The most common use case for offline transactions is through groups: even when using MOD, the initial group is broken down according to the hierarchy. The subordinate groups are then transferred to the Allevo planning transaction. A file is created for each of these groups.

The structure of the Excel file that is created depends on the execution type selected for the offline transaction. Two options are available:

- With the execution type “MO = MultiObject”, Allevo takes on the parameters from the selection in MOD view (including subordinate groups). The corresponding objects appear as a list on the Allevo sheet.
- With the execution type “MP = MultiPage” Allevo also takes on the transferred group – the group is then broken down as with typical MultiPage applications, and an Excel sheet is created for each object. This corresponds to the MOMP structure (as described in the previous section).

## 16.11 Optional: MOD expander (show/hide hierarchy levels)

When using MOD functions, multiple hierarchy levels are often used. In order to maintain a clear overview, it may be advantageous to restrict the display to a predefined hierarchy level with the option of expanding or collapsing subordinate levels by double-clicking on them (e.g. within the cost element structure).

|              |  |
|--------------|--|
| <b>Note:</b> | The double-click function is made available by additional VBA macros, that are currently not contained within the Allevo Standard Master. If interested in this function, please contact our support team. |
|--------------|--|

The function is controlled via an additional range (in this case, the range TE\_NAVSUMRNG): when double-clicking on a row in this range, the respective subordinate rows are displayed via standard Allevo Navigation functions. Upon a second double click, the rows will be hidden again.

## 16.12 Optional: MOD as cost allocation sheet (OAAS)

The list display of the MOD can be converted to a cost allocation sheet view using additional settings, which is based on an overhead allocation sheet: in this case, the values of the relevant objects (such as cost centers) or object groups are displayed in columns next to each other. This view is currently only available for cost element structure.

|              |   |
|--------------|---|
| <b>Note:</b> | This function is currently not contained in the Allevo Standard Master. If interested, please contact our support team. |
|--------------|---|

Here is an example:

**Allevo-MO: Kostenstelle**

Referenzdaten lesen | Satellit lesen | Satellit speichern | Einzelposten | Kommentare speichern | Kommentar-Übersicht | Status: In Bearbeitung

Datei: C:\temp\OAA5.xlsm

Navigation | Start | Einfügen | Seitenlayout | Formeln | Daten | Überprüfen | Ansicht | Entwicklertools | Allevo Maintenance

OaasFreeze

|     | A | G        | H                    | J | K         | L          | M          | N     | R         | S       | T          | U     | V               | W       | X        | Y     |
|-----|---|----------|----------------------|---|-----------|------------|------------|-------|-----------|---------|------------|-------|-----------------|---------|----------|-------|
| 4   |   |          |                      |   |           |            |            |       | H1110     |         |            |       | H1120           |         |          |       |
| 5   |   |          |                      |   |           |            |            |       | Vorstand  |         |            |       | Interne Dienste |         |          |       |
| 7   |   |          |                      |   |           |            |            |       |           |         |            |       |                 |         |          |       |
| 9   |   | CY_RH2_V | Plan1-12 2013        |   | Plan      | Plan       | abs        | %     | Plan      | Plan    | abs        | %     | Plan            | Plan    | abs      | %     |
| 10  |   |          |                      |   | 1-12 2013 | 1-12 2015  |            |       | 1-12      | 1-12    |            |       | 1-12            | 1-12    |          |       |
| 11  |   |          | Cost Element         |   |           |            |            |       |           |         |            |       |                 |         |          |       |
| 16  |   | 430000   | Salaries             |   | 2.325.814 | -2.325.814 | -100%      |       | 2.159.614 | 547.701 | -161192    | -75%  | 166.200         | 94.158  | -72.042  | -43%  |
| 50  |   | #        | Salaries             |   | 2.325.814 | 0          | -2.325.814 | -100% | 2.159.614 | 547.701 | -161192    | -75%  | 166.200         | 94.158  | -72.042  | -43%  |
| 55  |   | 435000   | Annual Bonus         |   | 95.416    | -95.416    | -100%      |       | 81.419    |         | -81.419    | -100% | 13.997          |         | -13.997  | -100% |
| 59  |   | 440100   | Soc secur., salary   |   | 466.349   | -466.349   | -100%      |       | 430.280   |         | -430.280   | -100% | 36.069          |         | -36.069  | -100% |
| 63  |   | 449000   | Other pers. costs    |   | 135.057   | -135.057   | -100%      |       | 123.334   | 5.792   | -117.542   | -95%  | 11.723          | 361     | -11.362  | -97%  |
| 64  |   | #        | Other Personal Costs |   | 696.822   | 0          | -696.822   | -100% | 635.033   | 5.792   | -629.241   | -99%  | 61.789          | 361     | -61.428  | -99%  |
| 65  |   | #        | Personnel Costs      |   | 3.022.635 | 0          | -3.022.635 | -100% | 2.794.646 | 553.493 | -2.241.153 | -80%  | 227.989         | 94.518  | -133.470 | -59%  |
| 66  |   | 400000   | Raw Materials 1      |   | 1.581     | -1.581     | -100%      |       |           |         |            |       | 1.581           |         | -1.581   | -100% |
| 81  |   | #        | Material Costs       |   | 1.581     | 0          | -1.581     | -100% | 0         | 0       | 0          |       | 1.581           | 0       | -1.581   | -100% |
| 83  |   | 481000   | Cost-acctg deprec    |   | 118.342   | -118.342   | -100%      |       | 8.241     | 5.121   | -3.120     | -38%  | 110.101         | 81.291  | -28.810  | -26%  |
| 87  |   | 483000   | Imputed interest     |   | 71.510    | -71.510    | -100%      |       | 5.970     | 3.137   | -2.833     | -47%  | 65.540          | 41.707  | -23.833  | -36%  |
| 90  |   | #        | Imputed Costs        |   | 189.852   | 0          | -189.852   | -100% | 14.211    | 8.258   | -5.953     | -42%  | 175.641         | 122.998 | -52.643  | -30%  |
| 96  |   | 416300   | Water                |   | 22.589    | -22.589    | -100%      |       |           |         |            |       | 22.589          | 6.936   | -15.653  | -69%  |
| 103 |   | 451000   | Building maintenance |   | 34.673    | -34.673    | -100%      |       |           |         |            |       | 34.673          | 14.367  | -20.306  | -59%  |
| 108 |   | #        | External Services    |   | 57.262    | 0          | -57.262    | -100% | 0         | 0       | 0          |       | 57.262          | 21.303  | -35.959  | -63%  |

Figure 16.7: Output example for MOD with cost allocation sheet structure (OAA5)

This display is referred to in Allevo as OAA5 (“Operating Accounting and Analyzing Sheet”), and in the following example, we are dealing with the OAA5 – Master. This display structure is especially helpful with Allevo Reporting. It has similarities to the Totals sheet in MultiPage-mode, but offers a number of advantages:

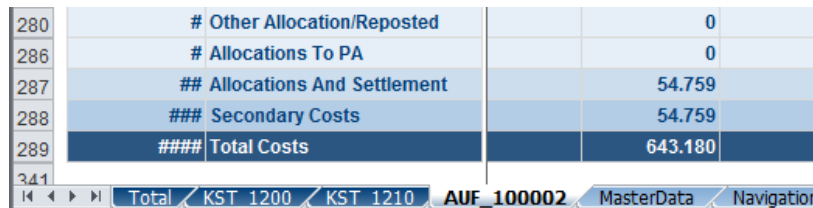
- No MultiPage sheets are created; this leads to performance improvements when reading data.
- Columns can contain either objects or groups; in the latter case, the constant MULTI\_WITH\_GROUPS must be set. This ensures that the corresponding ID code is interpreted as a group when reading data (Allevo then automatically reads the values for all objects within a group).
- The structure of the columns is variable: in the example above, two time ranges are compared to each other (plus calculated values for absolute deviation and percentage). The column layout is flexibly stored in the Allevo Master via ranges with sample displays. Columns with internal calculations have no header reference to Allevo column definitions for reading reference data.
- The user can later adjust which data is displayed in the individual columns (using the selection list for the available column definitions).
- The constant GRP\_LEVEL\_RESOLVE can be used to set the level of the display. For example, an entry of "1" means that only one level of the entry group should be broken down: on the Excel side, the values of the directly subordinate groups are always totaled (currently, the constant is only applicable with the sorting types 2,3,4 and 5).

**Special notes:**

- When calling via an Allevo-Reporting transaction, the list of relevant cost elements must be permanently stored (see constant DYN\_KSTAR\_FIX). A free selection using selection fields is not yet available.
- For the OAAS view, the MOD summation levels can sometimes be annoying and unnecessary. They can be completely hidden via constant DYN\_SAT\_CHAR; macro "StartSum" is not necessary.
- The OAAS Master can be conveniently accessed via the ABC, even in interaction with the tree: a group selected there is broken down and transferred to the OAAS view (constant OBJ\_SEL\_IN\_EXCEL).
- The constant GRP\_LEVEL\_RESOLVE allows you to call up evaluations, especially at the group level. In this case, it is advisable to deactivate the constant DYN\_KSTAR\_PRE for performance reasons (but should be checked in individual cases).

## 17 MultiObject with variable sheet assignments

In addition to the MOM variant of Allevo MultiObject (planning in list form), it is also possible to use a separate Excel worksheet (sometimes called MOWO) for each object in the MultiObject. Figure 17.1 below shows a simple example with two cost centers and one order.



|     |                               |         |
|-----|-------------------------------|---------|
| 280 | # Other Allocation/Reposted   | 0       |
| 286 | # Allocations To PA           | 0       |
| 287 | ## Allocations And Settlement | 54.759  |
| 288 | ### Secondary Costs           | 54.759  |
| 289 | #### Total Costs              | 643.180 |

Figure 17.1: Example MO with variable sheet assignment for the start cost center 1000

To use this variant of MO, two conditions must be met:

- In the settings on the |CUSTOMIZING| sheet, the setting “MOM without assigned objects” needs to be set to False (see also Chapter 10).
- There must be an assignment of objects to the start object as well as the assignment of a suitable template for the worksheet.

The relevant objects and corresponding worksheet templates are assigned via entries in the "Multi-Object with assigned objects" table on the "Masterdata" sheet. Here is an example:

MultiObject with assigned objects

| LeadingObject | AssignedObject | ObjectType | Template |
|---------------|----------------|------------|----------|
| 1000          | 1200           | KS         | Allevo   |
| 1000          | 1210           | KS         | Allevo   |
| 1000          | 100002         | OR         | Order    |
| 1110          | 1300           | KS         | Allevo   |
| 1110          | 1400           | KS         | Allevo   |
| 1110          | 100003         | OR         | Order    |

Figure 17.2: Assignment table for variable sheet assignment

Column meanings:

| Column         | Explanation  |
|----------------|--|
| LeadingObject  | Start object used to begin planning                                      |
| AssignedObject | The respectively assigned object   |
| ObjectType     | The object type of the assigned object                                   |
| Template       | Name of the Excel sheet to be used as a template for the assigned object |

You can work with copies of different spreadsheet templates. Usually, you create a spreadsheet template for each object type, which then takes into account the specific properties of the respective object type; however, it is also possible to enter a spreadsheet template directly at object level.

The entries in the example in Figure 17.2 show data that correspond to the data in Figure 17.1 above (starting via cost center MO): The representative cost center 1000 has two other cost centers assigned to it as well as an internal order.

In the example above, the table sheet template is actually selected at the object type level: however, there is no assignment directly to the object type in the table; therefore, entries must be made in each row. This slightly increased maintenance effort makes it possible to use table template sheets depending on the planning object.

|              |   |
|--------------|---|
| <b>Note:</b> | In the example, the assignments are permanently stored in the Master; the content of this table can also be made available by satellite 00. |
|--------------|---|

The sheets generated by Allevo follow a standard name assignment according to the following pattern (analogous to "IPP\_" for Multi):

- KS (= cost center): "KST\_" & AssignedObject
- OR (= order): "AUF\_" & AssignedObject
- PR (= Project structure plan): "PSP\_" & AssignedObject
- PC (= Profit center) "PRC\_" & AssignedObject (as of Allevo 3.4)
- KX (= Allevo object) "KER\_" & AssignedObject (as of Allevo 3.4)

Other names are possible via rules in the table "MOWONamingRule" on the sheet "MasterData". Allowed elements are: CC\_OBJECT, CC\_KOKRS, CC\_SETCL, CC\_VERSION, CC\_YEAR, CC\_PERITO (where CC\_OBJECT is displayed in external format, for example, without leading zeros for cost centers). All other rules are analogous to the functions in MultiPage mode.



## 18 Additional functions in the Allevo Master

### 18.1 Save any object relevant data in SAP

During planning, it can be helpful to manage any additional data for the object and to have it available the next time you call up planning (e.g., for auxiliary calculations). To make this possible, you can set up a range in the Allevo Master where the data is automatically stored in an SAP table and read from there at a later time. Data is transferred either with reference to the layout or only with reference to the object (but generally without reference to the year and/or version).

|              |   |
|--------------|---|
| <b>Note:</b> | Of course, a satellite could also be set up for this kind of task (if necessary, without reference to year and version). Disadvantage: the structure of a satellite table is largely fixed and must first be set up in the SAP system using an append. The solution here works completely without any preparatory work in the SAP system. |
|--------------|---|

On the Excel side, a structured table with the name "ZZObjectFields" must exist: Allevo takes all data in this table into account when writing and reading. The structured table must be created with a heading and this heading must not contain any spaces (to avoid error messages from Excel).

You can enter any data in the range of the structured table: there is no default in relation to the data type. The number of data records can also be changed at runtime; Allevo automatically adjusts the length of the table.

The table contents are transferred to SAP in XML format and stored in the database table /KERN/IPP-FIELDS. The contents cannot be evaluated on the SAP side.

|              |  |
|--------------|--|
| <b>Note:</b> | The constant NO_LAYOUT_FOR_FIELDS must be set if the data are to be saved without a reference to the layout. |
|--------------|--|

### 18.2 Change column properties

The Allevo Tree function allows the user to switch between objects from Excel. The function described here also allows you to change central characteristics that describe the contents of the displayed columns (such as year or version).

|              |  |
|--------------|--|
| <b>Note:</b> | This function was originally only intended for Allevo reporting mode (for example, when starting transaction /KERN/IPPKSREP for cost centers). It can also be used in planning, for example, to temporarily access actuals data from another year. Depending on the selection combinations, however, changing plan columns can be problematic (characteristic changes in CX_WW are ignored on the SAP side). |
|--------------|--|

If suitable parameters have been defined in Customizing for the Master (see below), the user can see the input fields with the characteristics of a column definition in the "Navigation" menu.

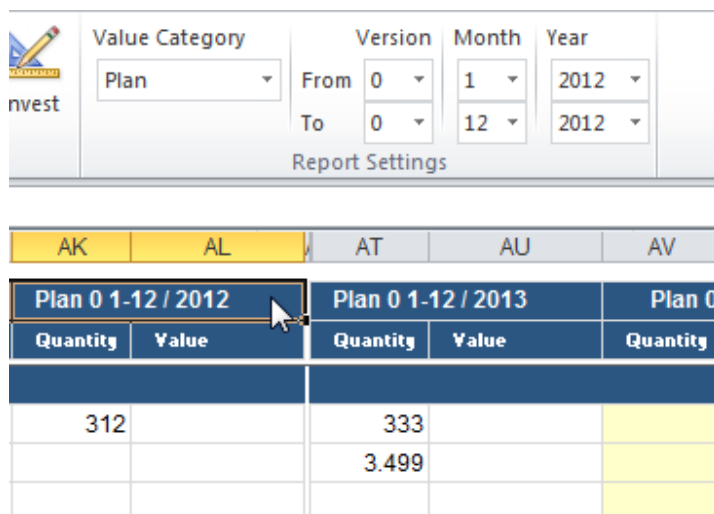


Figure 18.1: Change individual settings for a column definition

The contents of the fields offered always refer to the column that was previously selected via mouse-click (click on header cell). If an entry is changed (e.g. changed to another year), Allevo deletes the data in this column. In addition, this column is highlighted until reference data is imported again.

**Note:** To achieve the marking of the column, the relevant header cells must have "KernHeadlineParameter" saved as a format template (simply create a copy of KernHeadline1 and enter the desired color there).

The values that the user can select must be defined in Customizing using a table "ZZCustomizing-HeadlineSelection" (with header).

| ValueCategory | CategoryDescription | VersionFrom | VersionTo | PeriodFrom | PeriodTo | YearFrom | YearTo |
|---------------|---------------------|-------------|-----------|------------|----------|----------|--------|
| 1             | Ist                 | 0           | 0         | 1          | 6        | 2005     | 2005   |
| 2             | Plan                | 1           | 1         |            | 12       | 2006     | 2006   |
| 3             | Obligo              | 2           | 2         |            |          | 2007     | 2007   |
| 4             | Budget              |             |           |            |          | 2008     | 2008   |
|               |                     |             |           |            |          | 2009     | 2009   |

Figure 18.2: Parameters for column definitions: restricting the selectable values

The parameters of this table define which contents the user can change via the menu ribbon: in the example above, the entries for the year can be changed to values between 2005 and 2009. The table is interpreted column by column: in the example, the start period is always fixed at 1; in the case of a to-period, 6 or 12 can be entered.

The input fields shown above are available in the navigation menu only if this table exists. You can only select the columns that are available in the table and have a value (be sure to use correct spelling for column headings).

**Note:** On the SAP side, the constant DYNAMIC\_COLDEFS must also be set to active. This constant defines whether changes on the Excel side should be transferred (e.g., depending on the reporting mode, see F1 documentation for the constant).

Please also note the following restriction: on the SAP side, there can be no active relative reference to CX\_RR in the column definition.

### 18.3 Customer specific VBA extensions in the Allevo Master

Allevo projects are Excel projects: the flexibility of Excel makes it possible in many cases to set up a Master in such a way that the customer's desired design and functionality requirements are met (with the help of Excel formulas or Allevo navigation).

In some cases, it may be useful or even necessary to store customer specific functions as VBA coding in the Allevo Master (e.g., to depict special entry options during planning). Such solutions can often be called up like individual macros and in this way, they do not interfere with the other functions stored in the Allevo Master.

In other cases, it may be necessary to expand the base functions of the Allevo Master itself. Two examples for such a requirement:

- Individual checking of the completeness of data in the Allevo Master before calling an SAP function such as "Write data to SAP".
- Customer specific PDF output (see next section).

To enable such functions, the Allevo Master's VBA code contains "jump points" where customer-specific coding can be added (called "Cado" for short, similar to the user exits in the SAP system). These enhancements are usually carried out as part of an Allevo implementation project.

### 18.4 Optional VBA extensions in the Allevo Master

Since not all functionalities are necessary for each project, some modules are not standard in the Allevo Master. By leaving unnecessary components out, the Master is not overburdened with unnecessary code. As needed, the following modules can be simply inserted and integrated into the Master:

- CheckFormulaErrors
- FormulaFunction
- ModGenerator
- PickList
- PrintStatistics
- TotalSheet
- MOD-Expander

### 18.5 Export as PDF

In many use cases, an Allevo layout is also used for reporting purposes. In this case, only data is displayed and no plan values are entered. In such cases, a PDF output can also be useful.

Allevo supports PDF output on the SAP side, for example, when generating data in offline mode or directly in the planning screen via the "Save As" button (see also the documentation for the constant `BUTTON_SAVE_AS`).

In the Master, the PDF output must be prepared by means of corresponding individual output functions (e.g. specification of ranges to be output, similar to print functions). The output itself is done via

the macro 'Antloop.SaveAsPdf' (available as of version 3.2). Complex outputs (e.g., across several ranges of a sheet) can be created using customer-specific Excel macros.