

Peregrine  
**AssetCenter**



# Tailoring

© Copyright 2004 Peregrine Systems, Inc.  
All Rights Reserved.

Information contained in this document is proprietary to Peregrine Systems, Incorporated, and may be used or disclosed only with written permission from Peregrine Systems, Inc. This manual, or any part thereof, may not be reproduced without the prior written permission of Peregrine Systems, Inc. This document refers to numerous products by their trade names. In most, if not all, cases these designations are claimed as Trademarks or Registered Trademarks by their respective companies.

Peregrine Systems® and AssetCenter® are trademarks of Peregrine Systems, Inc. or its subsidiaries.

This document and the related software described in this manual are supplied under license or nondisclosure agreement and may be used or copied only in accordance with the terms of the agreement. The information in this document is subject to change without notice and does not represent a commitment on the part of Peregrine Systems, Inc. Contact Peregrine Systems, Inc., Customer Support to verify the date of the latest version of this document.

The names of companies and individuals used in the sample database and in examples in the manuals are fictitious and are intended to illustrate the use of the software. Any resemblance to actual companies or individuals, whether past or present, is purely coincidental.

If you need technical support for this product, or would like to request documentation for a product for which you are licensed, contact Peregrine Systems, Inc. Customer Support by email at [support@peregrine.com](mailto:support@peregrine.com).

If you have comments or suggestions about this documentation, contact Peregrine Systems, Inc. Technical Publications by email at [doc\\_comments@peregrine.com](mailto:doc_comments@peregrine.com).

This edition applies to version 4.4 of the licensed program

AssetCenter

Peregrine Systems, Inc.  
3611 Valley Centre Drive San Diego, CA 92130  
858.481.5000  
Fax 858.481.1751  
[www.peregrine.com](http://www.peregrine.com)



# Table of Contents

Introduction . . . . .	9
What is the aim of this guide? . . . . .	9
Who is this guide intended for? . . . . .	9
Contents of this guide . . . . .	10
<b>I. Customizing client computers . . . . .</b>	<b>13</b>
Chapter 1. Customizing a client workstation . . . . .	15
Activating the modules . . . . .	15
Customizing the toolbar . . . . .	16
Customizing tabs . . . . .	18
General options . . . . .	19
Customizing buttons . . . . .	21
User configurations . . . . .	23
Chapter 2. Using views . . . . .	25
Definition of a view . . . . .	26
To create a view . . . . .	27
To display a view . . . . .	27
To delete a view . . . . .	28
To modify a view . . . . .	28
To edit a view . . . . .	29
Associating a view with a button in a screen . . . . .	29

II. Customizing databases . . . . .	31
Chapter 3. Customizing the database . . . . .	33
Customizing existing objects . . . . .	33
Creating new objects . . . . .	44
Customization prepared in a test database . . . . .	63
Customization made by several operators at the same time . . . . .	68
Chapter 4. Database options . . . . .	95
To configure the database options . . . . .	95
Database option configuration window . . . . .	96
Indexes . . . . .	105



# List of Figures

1.1. Configuring tabs in a detail . . . . .	19
2.1. Items stored in a view . . . . .	26
3.1. AssetCenter Database Administrator - main screen . . . . .	34
3.2. Customization by several operators - simplified process . . . . .	71





# List of Tables

3.1. HTML tags recognized in help on fields and links . . . . .	41
3.2. Reserved characters . . . . .	41
3.3. Database customization - Tables to take into consideration . . . . .	87
4.1. Database options - Description of the available options . . . . .	97





# Introduction

---

## What is the aim of this guide?

This guide explains how to customize AssetCenter:

- At the level of the client workstation
- At the level of the database

---

## Who is this guide intended for?

This guide is intended for the following persons:

- AssetCenter end users wishing to customize the user interface
- AssetCenter administrators who wish to customize an AssetCenter database

# Contents of this guide

## Chapter Customizing a client workstation

This chapter explains how to customize the AssetCenter user interface at the level of the client machines:

- Modules activated
- Toolbar
- Tabs
- General options
- Buttons

## Chapter Using views

This chapter explains how to create, display, delete, or modify views and how to associate them with buttons.

## Chapter Customizing the database

This chapter explains how to customize an AssetCenter database:

- Customizing existing objects
- Creating new objects
- Preparing a customization in a test database
- Managing the tailoring process as a team

## Chapter Database options

This chapter explains how to configure the database options.

These options are common to all client computers connecting to the database.

# Conventions used in this guide

What follows is a list of the conventions that we use in this guide

Convention	Description
Java Script Code	Example of the code or command

Convention	Description
Fixed-width characters	DOS command, function parameter or data format
...	Portion of omitted code or command.
Note:	Informative note
Extra information	
<b>Important:</b>	Important information for the user
Be careful...	
<b>Tip:</b>	Tip to help you use the application
User tip	
<b>Warning:</b>	Extremely important information for the user
Exercise caution	
<b>Object</b>	AssetCenter interface object: menu, menu entry, tab or button.

The following conventions are also used:

- The steps that we ask you to follow in a specific order are shown as a numbered list. For example:
  - 1 First step
  - 2 Second step
  - 3 Third and last step
- All the figures and the tables are numbered according to the chapter in which they are found, and the order in which they appear in the chapter. For example, the title of the fourth table in chapter two will be prefixed by **Table 2-4**.





**PART**

# Customizing client computers





# 1 Customizing a client workstation

## CHAPTER

This chapter's objective is to teach you how to customize your workspace.

You will learn how to:

- Only select the functionalities that are useful to you.
- Customize the organization of information in your screens.
- Define the options that best adapt the interface to your needs.

All users can customize AssetCenter at the workstation level. These modifications only concern the workstation of the user who made them. Other workstations are not affected.

If you want to return to the default configuration for the screens (the windows describing tables), cancel your customization options for the duration of your session by selecting the **Tools/ Reinitialize screens** menu item.

---

## Activating the modules

AssetCenter enables you to restrict the active modules on a computer. Each module covers an area of functionality and brings menus, screens and wizards to the graphical interface.

By deactivating certain modules that you do not use, you simplify the user interface.

Use the **File/ Activate modules** menu item to select the modules to activate individually at the level of each client machine.

To enable to disable a module, use one of the following procedures:

- 1 Click the icon of the module.

---

 **Note:**

You can also cycle through the list of modules using the arrow keys and toggle them off or on using the spacebar.

- 2 Click **OK** to validate your modifications.
- 

## Customizing the toolbar

This section explains how to customize the AssetCenter toolbar.

### Adding a button to the toolbar

- 1 Use the **Tools/ Customize toolbar** menu item to configure the toolbar.



- 2 Select the tool category containing the button you want to add.

AssetCenter displays in the **Tools** frame the buttons corresponding to the selected category. The **Description** frame provides the description of the selected button.

- 3 In the **Tools** zone, select the button of your choice and drag it to the desired position on the toolbar.

AssetCenter integrates the button into the toolbar and resizes it as necessary.

## Removing a button from the toolbar

Select the button and drag it outside the toolbar. AssetCenter modifies and resizes the toolbar as necessary.

## Repositioning a button in the toolbar

Select the button and drag it to another position in the toolbar. AssetCenter modifies and resizes the button as necessary.

## Adding a separator in the toolbar

Select the button to the right of (or below) the area where you want to add a separator. Then, drag it to its own right half (lower half if the toolbar is in the vertical position). AssetCenter adds a separator in front of the button and resizes the toolbar if necessary.

---

 **Note:**

It is not possible to have two separators together; you can only have one between two buttons.

---

## Removing a separator from the toolbar

Select the button immediately preceding the separator and drag it over the separator. AssetCenter deletes the separator and resizes the toolbar if necessary.

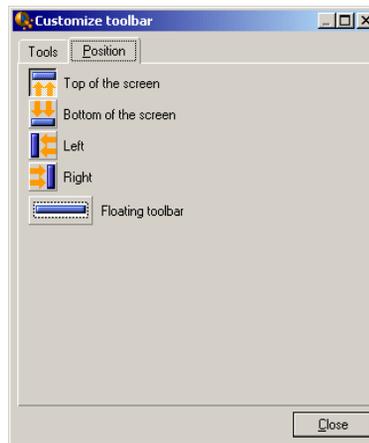
## Displaying the default toolbar

AssetCenter is delivered with a standard toolbar that you can restore by selecting the **Standard toolbar** checkbox.

If you restore the standard toolbar, any customizations are lost.

## Modifying the position of the toolbar

The position of the toolbar in the workspace can be modified using the **Position** tab in the toolbar customization window.



Click the button corresponding to the position in which you want to find your toolbar. Or, you can move the toolbar anywhere you like on the workspace by choosing **Floating toolbar**.

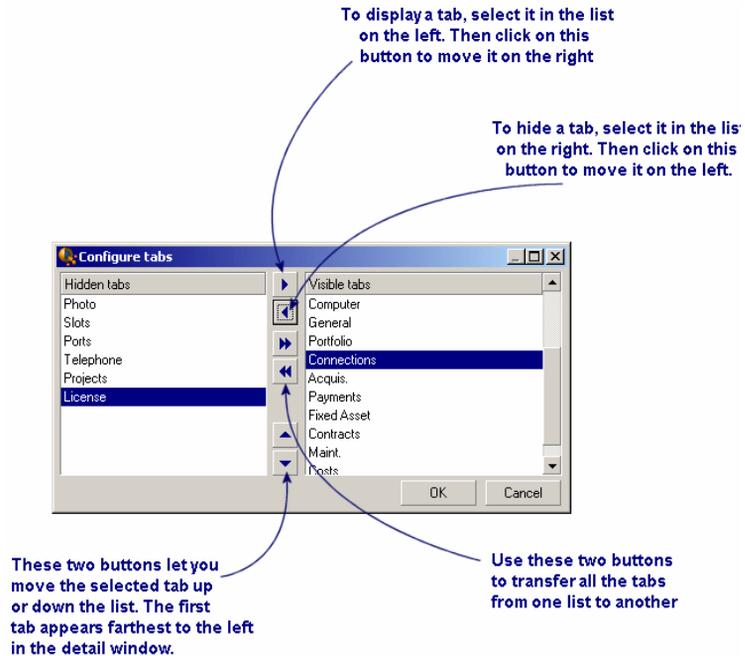
---

## Customizing tabs

Each user can select the list of tabs to be displayed in a detail window and the order in which they are displayed. Other workstations are not affected by these modifications.

Use the **Tools/ Configure tabs** shortcut menu or **Tools/ Configure tabs** from the general menu when your cursor is in the tab zone to make these modifications.

Figure 1.1. Configuring tabs in a detail



- To show a hidden tab, you can double-click on its name in the list on the left.
- To hide a visible tab, double-click on its name in the list on the right.
- You can change the order of items in the right-hand list. Click on the desired item and use the up and down arrows to the right of the list to move it to a new position. The tab at the top of the list appears on the left-most side of the detail window.

## General options

AssetCenter's options can be configured via the **Edit/ Options** menu.

---

 **Note:**

These options are either associated with the computer on which they are defined or with the user (and therefore their login) who defined them during a work session. The options that apply to the workstation are available without the user having to be connected.

---

To find out what an option corresponds to and how to configure it:

- 1 Display the options window (**Edit/ Options** menu).
- 2 Select the option in question.
- 3 Consult the **Description** field, which is also in the options window.

To modify an option:

- 1 Display the options window (**Edit/ Options** menu).
  - 2 Select the option to modify.
  - 3 Click in the **Value** column.
  - 4 Select a value from the list (if a list is available) or enter a value manually.
- 

 **Note:**

The modifications made to the options in this window are saved after you click **OK**.

---

## Functions of the buttons

- **Reset:** Sets all options to their default values and saves.
- 

 **Warning:**

You cannot cancel the action once you reset.  
You cannot reset a sub-set of options.

---

- **Apply:** Saves the values of all the options but does not close the window.
- 

 **Warning:**

You cannot cancel this action once you apply.

---

- **OK:** Saves the values of all the options and closes the window.
-

## Administrator information

There are several types of options:

- The options independent of the databases.  
Their values are stored on the hard drive of the machine where these options are defined.
- The options dependant on a database.  
Their values are stored in the database and associated with the login of the person making the modifications.  
Anyone who connects with the same login, even if it's from a different computer, will find the same optional settings.



Tip:

To find out whether an option is independent on a database:

- 1 Launch AssetCenter without connecting to a database (or if you are already connected, disconnect from the database).
  - 2 Display the options window (**Edit/ Options** menu).
  - 3 The only options available in this window now are those that are independent of the database (and stored on the hard drive of the computer). All database-dependent options in this window are grayed out.
- 

## Customizing buttons

Buttons in screens may be associated with any of the following object types:

- Actions
- Screens
- Views
- Forms
- Reports

The main function of the button remains enabled (for example, the **Duplicate** button still duplicates records), except for the ,  and  buttons, which once associated with an object lose their main function.

The associated object is associated:

- Before the main functions of the **New** and **Duplicate** buttons.
- After the main functions of the **Creat**, **Modify** and **Create/ Duplicate** buttons.

In certain cases, if the associate object does not execute successfully, the main function of the button is canceled.

## Prerequisites

To modify the buttons and associate an object to them:

- 1 Display the options window (**Edit/ Options**).
- 2 Select the option **Edit/ Enable customization of buttons in screens**.
- 3 Set this value to **Yes**.
- 4 Click **OK**.

## Associating an object with a button

To associate an object with a button:

- 1 Display the screen to which the button belongs (**Portfolio/ Computers**, for example).
- 2 Right-click the button to modify, and then select **Customize** from the shortcut menu.
- 3 Select the object to associate.
- 4 Click **OK**.

## Modifying the object associated with a button

To modify the object the object associated with a button:

- 1 Display the screen to which the button belongs (**Portfolio/ Computers**, for example).
- 2 Right-click the button to modify, and then select **Edit customization** from the shortcut menu.
- 3 Select the new object to associate.
- 4 Click **OK**.

## Remove the object associated with a button

To remove the object associated with a button:

- 1 Display the screen to which the button belongs (**Portfolio/ Computers**, for example).
- 2 Right-click the button to modify, and then select **Remove customization** from the shortcut menu.

**Warning:**

No confirmation is requested.

---

## Additional information

The modifications carried out on a button are immediately saved in the database. You do not need to use the **Administration/ Save the database configuration** menu.

The objects that are shown in the customization window are sorted by functional domain.

AssetCenter Database Administrator also enables you to:

- Add new buttons to screens
- Associate an object with a button

For further information, refer to:

- This guide, section [Buttons](#) [page 42].
- This guidel, section [Creating action buttons](#) [page 49].

---

## User configurations

By default, AssetCenter stores the user preferences (also called user configurations) in the **amSysConfig** table. When a user modifies their preferences (options, window sizes, enabled modules, list and sub-list configurations, filters and views, etc.), the corresponding record is updated in the database.

To stop users from updating their configurations and define a default configuration:

- Create a user configuration that suits you by customizing AssetCenter. You will use this configuration as the standard configuration.
- Select the **Administration/ List of screens** menu.
- Select the **amSysConfig** table.

- Select and duplicate all records whose **Section** starts with **User**.
- 

 **Note:**

The purpose of performing this duplication is to keep the original user configuration.

---

- Create an action that sets the value of the **EmployeeId** in the **amSysConfig** table to 0. This step enables you to specify that that configuration is applied to **all** users. To do this:
  - Select the **Tools/ Actions/ Edit** menu.
  - Click **New**, define a **Name** for the action (for example, **Config**), select **amSysConfig** as the **Context** and **Script** as the **Type**.
  - In the **Script** tab, enter the following script:

```
Set [lEmp1DeptId]=0
```

- Finish the creation by clicking **Create**.
  - Execute this action on the records duplicated previously.
  - Define for all users, a read-only user right for the **amSysConfig** table.
- 

 **Note:**

For further information defining user rights, refer to the **Administration** guide, chapter **Controlling access to the database**.

---

 **Important:**

This procedure is only valid for users that have not yet connected to the database, because they do not yet have a record in the **amSysConfig** table. If a user has already connected to the database, their preferences are stored in the table. The only way to cancel them and to force the default configuration is to delete the records corresponding to this user.

---

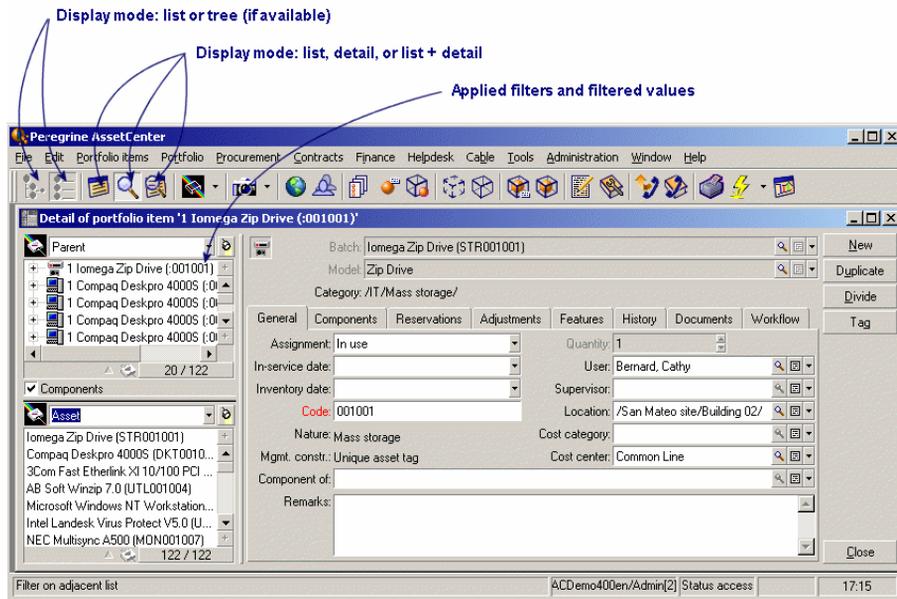


# 2 | Using views

CHAPTER

This chapter explains how to use views with AssetCenter.  
Use the **Tools/ Views** menu item to access the list of views.

Figure 2.1. Items stored in a view



## Definition of a view

A view lets you save the display parameters of a main list:

- The display mode: Table view or Tree view.
- Display mode: Details only, List only, or List + Details.
- Sort conditions.
- Filter applied, and the filter values.
- List of visible columns and their width.
- Dimensions of the window.

Creating a view makes it easier to produce the same report every time. Once you have displayed the view you can easily print it with the same presentation each time.

### Examples

- List of unassigned assets worth over \$5,000.

- List of all offices with a fax machine, displaying the office number, the fax model and the name of their users.

---

## To create a view

- 1 To create a view, display the list as you want it to be memorized.
  - 2 Save this view via the **Tools/ Views/ Create** menu from the current window.
- Any AssetCenter user with access to the table of views can create a view. The view's author decides whether the view should be available to other users by selecting or clearing the **Not shared** option. An AssetCenter administrator may display, modify and delete views made by all users, including the ones users decide not to make public.

---

## To display a view

Select the view among the names of views displayed in the **Tools/ Views** menu. The views available are those you have created and those made public by the administrator, as long as the **Visible in the menus** option (SQL name: bVisible) has been checked.

The view is displayed using all the saved parameters.

AssetCenter displays a new window, even if a list with the same information is already displayed. The window title displays the name of the view.

A view may be linked to a query wizard used to filter the list of records exposed in a view before being displayed.

For further information on query wizards, refer to the **User Interface** guide, chapter **Searching for items in a table**, section **Query wizard (QBE)**.

---

 **Note:**

You can also use the  button in the toolbar to display the list of views. This button displays the icon of the view that was last called.

---

---

## To delete a view

- 1st method:
  - 1 Display the view.
  - 2 Then delete it using the **Tools/ View/ Drop current view** menu item.
- 2nd method:
  - 1 Select the **Tools/ Views/ Edit** menu item to display the table of views.
  - 2 Select the view you want to drop.
  - 3 Click **Delete**.

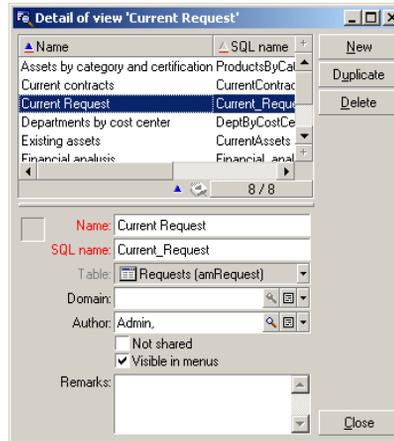
---

## To modify a view

- 1 Display the view on screen.
- 2 Modify your display parameters.
- 3 Save your modifications using the **Tools/ View/ Modify current view** menu item.

## To edit a view

Using the **Tools/ Views/ Edit** menu item, the administrator can access the list of all existing views:



The administrator must assign a value to each one of the fields in the detail of the selected view.

## Associating a view with a button in a screen

There are several ways to associate a view with a screen button.

For further information, refer to:

- This guide, section [Customizing buttons](#) [page 21].
- This guide, section [Buttons](#) [page 42].
- This guidel, section [Creating action buttons](#) [page 49].





# Customizing databases





# 3 Customizing the database

## CHAPTER

AssetCenter Database Administrator enables you to customize the database in order to tailor it to the needs of your enterprise.

This level of customization is restricted to an administrator.

Modifications are visible to AssetCenter users. All users view the database in the same way, as defined by the administrator. When the administrator modifies the name of a field or a link, the new name appears whenever it is used in AssetCenter, in particular in the list screens, detail screens, and when building queries.

The following sections provide a detailed explanation of how to customize the various objects in the AssetCenter database (tables, fields, links, indexes, screens, etc.):

- Existing objects.
- New objects.

---

## Customizing existing objects

AssetCenter Database Administrator authorizes only limited customizations of the existing database objects. In order to avoid any problems while using AssetCenter, certain object values and objects are in read-only. Typically, you cannot modify the pages of an existing detail.

To customize an object in the database, edit the database description file (**gbbase.xml**).

Two distinct forms of database customization are handled by AssetCenter Database Administrator:

- Customization before creating the database.
- Customization after creating the database.

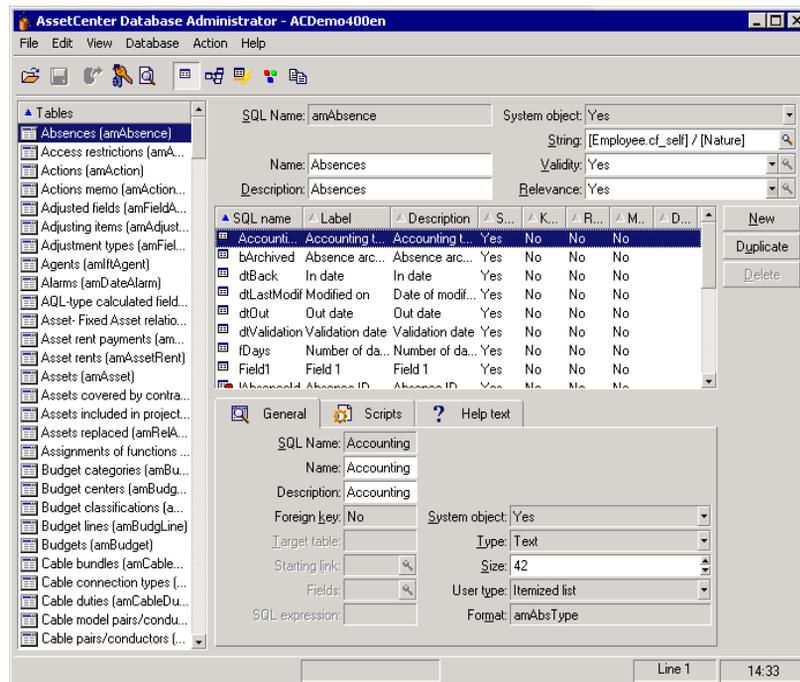
 **Note:**

The difference between these two cases is that you can only modify the size of "Text" fields before creating the database.

For each of these two cases, two levels of customization are possible:

- Customizing a table.
- Customizing objects (fields, links, indexes, screens, pages) of a table.

Figure 3.1. AssetCenter Database Administrator - main screen



## Customizing tables

To customize the table, you can modify:

- The **Description** field.
- The **Label** field, which contains the name of the table as it is displayed in AssetCenter.
- The **String** field, which enables you to build the string which represents a record from this table in AssetCenter.
- The **Validity** field applies to all records of a table in the database. It enables you to define the conditions of validity of creation or modification of a record in the table.
  - If this field is set to **Yes**, a record in the table can always be created or modified.
  - If this field is set to **No**, a record in the table can never be created or modified.
  - If this field is set to **Script**, you can define a script conditioning the validity of creation or modification of a record in the table.

For example, for **Numerical** type features, it is possible to restrict the creation of a feature if its value is outside certain limits. The following script concerns the "Validity" field in the "amFeature" table:

```
if [seDataType] = 1 and [fMin] > [fMax] Then
Err.Raise(-1, "The value of the 'minimum' field must be less than th
e value of the 'maximum' field.")
RetVal = FALSE
Else
RetVal = TRUE
End If
```

### Note:

When creation or modification of a record is invalidated by the value of the "Validity" field for the table concerned, it is good practice to raise an explicit error message via the standard Basic function "Err.Raise" in order to warn the user. If you do not do this, the user will not necessarily be able to understand why it is not possible to modify or create the record.

The "SQL name" field cannot be modified; it contains the table's SQL name.

- The **Can contain features** option enables you to include features in a table. This operation is irreversible.

## Customizing objects

The second part of this customization pane enables you to list all objects of a given type.

When you select an object in the list, AssetCenter Database Administrator displays the description of this object in the third part of the customization pane.

---

### Note:

The red  icon identifies the primary key of the table.

The yellow  icon identifies a foreign key.

The  icon identifies a field belonging to an index.

The  icon means that the indexed list of fields must not occur more than once in the table.

The  icon means that the indexed list of fields must not occur more than once in the table, except for the "NULL" value, which may be repeated any number of times.

The  icon represents a 1 link.

The  icon represents an n link.

---

## Customizing a field, a link or an index

### Customization with AssetCenter Database Administrator

For this type of object, you can modify:

In the **General** tab

- The **Name** field.
- The **Description** field.
- The **Size** field enables you to specify the size of "Text" type fields. It can only be accessed when customizing a database before creating it or when modifying a database description file. The maximum size is 255 characters.

And in the **Scripts** tab:

---

### Note:

Except where specified, the following attributes are taken into account both by the user interface and external tools when accessing the database.

---

- The **History** field: indicates whether history is kept for modifications made to the field and under what conditions (using a script).
- The **Read only** field: indicates whether it is possible to modify the field via the user interface and under what possible conditions (using a script).

 **Note:**

This attribute is not taken into account when importing data using external tools. The import module does not take these into account either, as long as the fields are not defined as "Read only" at the level of mappings. On the other hand, if an import script maps a source field to a read-only target field, the read-only field can be modified all the same.

- The **Irrelevance** field conditions the display of the selected record in the detail of a list.
  - **Yes:** deletes the display of the object.
  - **No:** conserves the display of the object.
  - **Script:** indicates that a script is attached to the object to specify its display conditions.

```
If [bDepartment] = 1 Then
RetVal = 1
Else
RetVal = 0
End If
```

The selected object is not displayed if it is not relevant.

- The **Filter** field is not used in this version of AssetCenter.
- The **Formatting** field: enables you to format the value of the field automatically before storing it in the database.

For text fields:

- **Standard:** stores the value as entered.
- **Uppercase:** converts the value to uppercase before storing it.
- **Lowercase:** converts the value to lowercase before storing it.
- **Automatic:** converts the first letter to uppercase before storing the value.

For numeric fields:

- **Standard:** accepts all numbers, both positive and negative.
- **Positive:** rejects negative values. The warning message is displayed.

 **Note:**

If you modify this attribute, existing values in the database will not be converted.

- The **Mandatory** field enables you to define conditions making a field mandatory.

 **Note:**

Making a field mandatory can pose problems if it is not always visible (for example, if its being displayed is determined by the value of a field). Always bear this in mind when configuring a field or writing a script.

- The **Default** field specifies the default value of a field.  
Default values for fields and links are built from a series of:

 **Note:**

AssetCenter automatically applies default values when creating a new record. Users creating or modifying the record can modify these values.

Default values are only applied when a record is created.

For example, if the default value of field x in record A refers to field y in record B, and if field y or record B is modified, the value stored in field x is unaffected unless record A is created after the modification.

- Constant values surrounded by "quotation marks".
- Script functions returning values.
- References to other database fields.

**Example:**

```
RetVal="BL"+AmCounter("Delivery", 2)+AmDate()
```

- AmDate() provides the date on which the record was created.
- AmCounter("Delivery, 2) provides the current value of the "Delivery" counter incremented by 1, and expressed in two digits.

In the **Help text** tab, you can customize the extended help concerning a database object. This help is available in AssetCenter via "Shift+F1" (or the **Help/ Help on this field** shortcut menu) and can include up to three sections. By default, these sections are entitled "Description", "Example" and "Precautions". The titles of these sections can be customized by modifying the labels of the links with SQL names "Comment", "Example" and "Warning" in the **Help on fields** table (amHelp).



Note:

The other fields in this section of the database customization screen are shown for informational purposes only and cannot be customized. Configuring the objects in this screen works in the same way as is available via the **Configure object** pop-up menu command.

---

## Default values for fields and links - additional information

### Calculated fields:

Calculated fields can only be used to calculate the default values of standard fields if their type is **Calculated string** or **Basic script**.

### Links to the Comments (amComment) table:

It is not possible to assign a default value to the links going to this table.

### Access restrictions:

The calculation of default values for fields does not take access restrictions into account. You must therefore only reference those fields and links that may be viewed by all users.

### Counters:

When you define a default value for a field, you can reference a counter.

When creating a new record containing this field, AssetCenter automatically inserts the number, incremented by one unit, for each new record.

Counters are managed by the administrator using the **Administration/ Counters** menu.

You must first create a counter using this menu before you can insert the counter in a formula for a field default value. Counter names cannot contain the following characters: space, "\$", "(" or "."

You can create as many counters as you want.

AssetCenter users never see the counter names.

The counter is incremented as soon as you click **New**. If you cancel the record creation without clicking **Create**, the counter is not decremented.

You are free to readjust the values of the counters.

The value displayed in the counter detail is the last number used for this counter.

---

 **Note:**

If you enter a formula such as AmCounter(<Counter name>, [n]) in the default value of a field instead of AmCounter(<Counter name>), the value of the counter will be displayed using n digits.

---

**Errors:**

Here are some of the most common mistakes made when describing default values:

Execution error - Type error

In some cases the variable may be incompatible with the field type. This is the case if you try to define AmLoginName() on a date-type field, for example.

Field 'XXX' unknown in 'XXX'

When you reference a link, you must use the "Link.Link.Field" format with each link is referenced by name. This makes it possible to follow the links directly.

For example, to set a default supervisor for an asset, you may replace the cost center supervisor with: "CostCenter.Supervisor".

The default value will be used as soon as the AssetCenter user populates the asset's cost center. Then the supervisor can be located in the database.

## Help on fields and links

The extended help system for AssetCenter fields uses a sub-set of the HTML language. Only a few "tags" are recognized, used mainly for formatting ToolTips. This section does not attempt to provide an exhaustive description of the HTML language. It simply gives a brief description of each of the HTML tags managed by AssetCenter.

---

 **Note:**

In the following table, the HTML tags are listed as they must be entered by the user. They do not follow the notation conventions applicable to the rest of the online help.

---

Table 3.1. HTML tags recognized in help on fields and links

HTML Tag	Description
<FONT FACE="font name">	Defines the font used to display the text following this tag. This font remains in use until the next change of font.
<FONT COLOR=#RRGGBB>	Defines the font color used to display the text following this tag. This color remains in use until the next change of color.
<FONT SIZE=+n>	Increases the font size by "n" levels. This font size remains in use until the next change of font size.
<FONT SIZE=-n>	Decreases the font size by "n" levels. This font size remains in use until the next change of font size.
<B> </B>	These two tags surround text to be displayed in bold.
<I> </I>	These two tags surround text to be displayed in italics.
<LI>	This tag is the start of a bulleted list. It causes a carriage return and inserts a bullet at the start of the text.
<HR>	Draws a horizontal line used as a text separator.

For further information about the HTML language, refer to one of the numerous publications on the subject.

#### Reserved characters:

The following table lists the reserved characters in the HTML language. If you enter these characters, they will not be displayed as such on the screen. You should use the corresponding tag to display the desired character.

Table 3.2. Reserved characters

Reserved character	Use the following tag to display this character
"<"	"&lt;"
"&"	"&amp;"
Non breaking space	"&nbsp;"

## Customization with AssetCenter

You can customize database fields and links using the **Configure object** command from the shortcut menu.

To display the **Configure object** shortcut menu, move to the field to configure and right-click.

---

### Note:

If you use the shortcut menu, changes are recorded: either when you close the database and confirm the validation message, or when you select the **Administration/ Save database configuration** menu.

---

You can also modify the context-sensitive help for a field or a link from the **Help text** tab:

- "Description": explains the contents of the field or link.
- "Example": provides examples of how to populate the field or link.
- "Precautions": lists "tricky" aspects to bear in mind: Data-entry precautions, linked automatic mechanisms, etc.

The help system for fields is a sub-set of the HTML language.

## Customizing a detail

For this type of object, you can modify:

In the **General** tab

- ◆ The **Name** field.

In the **List/Detail** tab

- List/Detail proportions
- Title of the detail
- Title of the list
- System filter
- (Functional) Domain
- Columns in the list

## Buttons

The **Buttons** tab enables you create, modify, delete and to organize the buttons in screens.

Buttons can be associated with any of the following object types:

- Actions

- Screens
- Views
- Forms
- Reports

## To create a button:

- 1 Click .
- 2 A line is added to the list of buttons exposed in the tab. Click on each of the line's cells to define the following button properties:
  - **Name:** internal name of the button. It enables you to identify the button in a unique manner.
  - **Text:** the text appearing on the button.
  - **Description:** label of the button as it is displayed in the graphical interface of AssetCenter.
  - **Multiple selection:** Enables you to specify if the actions taken when you click the button can apply to several records.
  - **No selection:** Enables you to specify if the actions taken when you click the button can be launched without any record having been selected.
  - **Associated action:** Enables you to define the action executed when you click the button. Enter the SQL name of the action to execute.

```
<Type of action>:<SQL name of the action, of the view, etc.>
```

In this syntax, the type of action can take the following values:

- **A** for an action.
- **S** for a screen.
- **V** for a view.
- **F** for a form.
- **R** for a report.
- **Relevance:** enables you to specify whether the button is visible, invisible or whether its visibility is conditioned by a script.

## To modify a button

- 1 Select the line corresponding to the button.
- 2 Click each cell to define the properties of the button.

If the button is a standard button, the primary function remains active (for example, the **Duplicate** button will always duplicate records), except for the

following buttons: ,  and . These buttons can still be associated with an object but their primary function becomes inactive.

The associated object is associated:

- Before the main functions of the **New** and **Duplicate** buttons.
- After the main functions of the **Create**, **Modify** and **Create/ Duplicate** buttons.

In certain cases, if the associate object does not execute successfully, the main function of the button is canceled.

## To delete a button

- 1 Select the button in the list displayed in the **Buttons** tab.
- 2 Click .
- 3 Click **Modify**.

## To modify the order of buttons

To modify the order in which the buttons appear in the detail, use the  and  buttons.

---

# Creating new objects

AssetCenter Database Administrator enables you to create new objects for the database.

---

### **Warning:**

We recommend that you work on an empty database. If you modify your working database, however, you must verify and repair the database (**Action/ Diagnostics/ Repair database** menu) after having saved your changes.

---

## Methods of creation

The following steps propose a method of creating new objects. These steps assume that you are creating the largest object: a table. Each step corresponds to a particular selection in this chapter.

- Create your table.

- Create the fields, links and index of your table.
- Create the details for your table.
- Create, if you want, action buttons present in the detail.
- Create the pages of your details.
- Add the pages to your details.
- Save your modifications.
- If this fails, propagate your modifications.

## Creating a table

To create a new table:

- Select the **Database/Add a table** menu.
  - AssetCenter Database Administrator displays a creation window.
  - In this window, populate the standard fields associated with a table:
    - The **SQL name** field enables you to identify the new table in a unique manner and, notably, to reference it in a Basic script.
    - The **Label** field contains the name of the table as it is shown in AssetCenter.
    - The **Description** field.
    - The **Primary key** field contains the SQL name of the field used as the table's primary key.
    - If you want to associate features to your new table, select the **Add features** option. AssetCenter Database Administrator will automatically create the additional tables necessary to deal with the features.
  - Click **Create**. AssetCenter creates the table as well as the field defined as the primary key for this table, and it puts you in position to edit the fields for this table.
- **Administration** guide, chapter **Standard database description files**, section **Description of tables**.

## Creating a field, link or index

To create a field:

- 1 Select the **Database/Add a field** menu.
- 2 AssetCenter Database Administrator displays a creation window.
- 3 In this window, populate the following fields:
  - **SQL name**

- **Name**
  - **Description**
  - **Type**
- 4 Select the **Create an index for this field** option if you want to create an index. After the field is created, you can no longer modify its size or format.
- 



Note:

Different types of fields:

- An **custom itemized list** type field must respect the following syntax:

```
Free text
```

The text entered (**Format** field) corresponds to the **Identifier** field in the detail of the itemized list.

- A **system-itemized list** type field must respect the following syntax:

```
To be validated|0|Validate|1|Refused|2
```

- A **Duration** type field must respect the following syntax:

```
%Yl%Ml%Dld%H%N%S
```

- **Y, M, D, H, N, S** define the type of duration: respectively, year, month, day, hour, minute, second.
- **l** (long) defines whether the displayed duration unit is complete or not.
- **d** defines whether the duration is the default duration.

---

► **Administration** guide, chapter **Standard database description files**, section **Description of fields**.

To create a link:

- 1 Select the **Database/Add a link** menu.
- 2 AssetCenter Database Administrator displays a creation window.
- 3 In this window, populate the following fields:
  - **Link type**
  - For the source table:
    - **SQL name**
    - **Name**
    - **Description**
  - For the destination table:
    - **Table**: Indicate the destination table.
    - **SQL name**

- **Name**
- **Description**

- 4 Select the **Create an index for this link** option if you want to create an index.
    - ▶ **Administration** guide, chapter **Standard database description files**, section **Description of links**.
- 

 **Note:**

Different types of links:

- The 1-N link types are the following: Normal, Define, Own, OwnCopy, Overflow
  - The following are the types of 1-1 links: Normal1, OwnCopy1, Copy1, Own1, NeutDef1
  - The following are the types of N-N links: ReINN, NNCopy, CopyNN
- 

To create an index:

- 1 Select the menu **Database/Add an index**.
- 2 Populate the different properties of each object. These properties are described in an exhaustive manner in the section **Customizing objects**, and are thus not detailed here.

## Creating a hierarchical table

Creating a hierarchical table means creating a parent link in that table.

A parent link corresponds to a child link in the same table.

To create a parent link:

- 1 Select the **Database/Add a parent link** menu.
- 2 AssetCenter Database Administrator displays a creation window.
- 3 In this window, populate the following fields:
  - **SQL name**
  - **Name**
  - **Description**
- 4 Populate the **Hierarchy** field according to the following syntax:

```
<Id of the parent link>,<name of the field to be used>
```

For example, for the Products table, the **Hierarchy** field of the **Parent** link is populated as follows:

```
1ParentId, InternalRef
```

After these steps are completed, a **FullName** field is created. This is a unique identifier inside that table.

► **Administration** guide, chapter **Standard database description files**, section **Description of indexes**.

## Creating a detail

A detail is a graphical view of the information stored in the objects of a table. The screens displayed in AssetCenter when you select a menu, click a toolbar icon or choose a view are all examples of a detail. Details are composed of several pages that are regrouped by tabs in the graphical interface of AssetCenter.

---

 **Note:**

AssetCenter is provided with a set of standard details that are used to support the functions of the application. These details cannot be edited or modified. If you want to add a page to an existing detail, for example, you need to duplicate the detail, name it, then add the desired items.

---

To create a detail:

- 1 Select in the left-hand pane the table for which you want to create a new detail.
- 2 Select the View/Details menu or click the  button on the toolbar.
- 3 Select the **Database/Add a detail** menu or click **New**.
- 4 Populate the following fields in the **General** tab:
  - SQL name: SQL name of the detail. This name enables you to identify the detail in a unique manner and to reference it in scripts or queries.
  - Name: name of the detail.
- 5 Populate the following fields in the **List/Detail** tab:
  - Title of the list: defines the title of the detail as it appears in the list of screens (**Administration/ List of screens**, in AssetCenter).
  - Title of detail: defines the title of the window displayed in AssetCenter. The **String** of the description is added to this title.
  - Domain: functional domain of this detail. If the functional domain that you enter already exists in the database (created using the **Administration/Functional domains** menu), you will automatically have access to your screen in the functions pane under the corresponding functional domain.

- List/detail proportions: ratio between the detail and the list.
- Columns of the list: This field enables you to define the columns that will be displayed in a list. The syntax of this field is the following:

```
<SQL name of the field for the column 1>,<Proportional size of the column>,...
```

## 6 Click **Create**.

You just created the empty shell of your new screen. Now all you need to do is populate this screen by adding pages that contain fields or links, and possibly action buttons.

## Creating action buttons

AssetCenter Database Administrator offers you to possibility of creating buttons in your details. These buttons enable you to trigger the execution of an action, the display of a screen, the printing of a report or form, or to open a view.

To create this button:

- 1 Select the detail to which you want to add a button.
- 2 Select the **Buttons** tab of the detail.
- 3 Click . A new item is created in the list of buttons for this tab.
- 4 You can directly edit the values for each property by clicking its value.
  - **Name:** SQL name of the button. This name enables you to identify the detail in a unique manner and to reference it in Basic scripts or queries.
  - **Text:** contains the button's text.
  - **Description:** label of the button as it is displayed in the graphical interface of AssetCenter when the cursor is over the button.
  - **Multi-selection:** enables you to specify if the actions taken when you click the button can apply to several records.
  - **Edit mode:** enables you to specify whether or not the button appears while saving an item.
  - **No selection:** enables you to specify if the actions taken when you click the button can be launched without any record having been selected.
  - **Associated action:** Enables you to define the action executed when you click the button. Enter the SQL name of the action to execute.

```
<Type of action>:<SQL name of the action, of the view, etc.>
```

In this syntax, the type of action can take the following values:

- **A** for an action.

- **S** for a screen.
  - **V** for a view.
  - **F** for a form.
  - **R** for a report.
  - **Relevance**: enables you to specify the button's display parameters (yes, no or script).
- 5 Click **Modify** to validate your changes.

## Creating a page

In AssetCenter, a page is graphically represented by a tab containing fields. To create this page:

- 1 Select in the left-hand pane the table for which you want to create a new page.
- 2 Select the **View/Pages** or click the  button on the toolbar.
- 3 Select the **Database/Add a page** menu or click **New**.
- 4 Populate the following fields in the **General** tab:
  - **SQL name**: SQL name of the page. This name enables you to identify the page in a unique manner and to reference it in scripts or queries.
  - **Name**: name of the page. This name appears as the name of the tab in the AssetCenter interface.
- 5 Click **Create**.
- 6 Click the **Contents** tab of the page's detail. You can now select the fields that will be available in the page that you are created. To do this:
  - Select in the list to the left (**Fields and links**) the field or the link that you want to add to the page, and click the  button to transfer it to the list on the right (**List of fields**).To perform the same operation, you can also:
  - Double-click the field in the list on the left (**Fields and links**).
  - Enter the SQL name of the field or link in the field above the list on the left and click the  button next to this list.

 Note:

Entering the information this way enables you to specify the type of control associated with the object and how it behaves by using keywords. These keywords are described later in this document.

- If you want to delete a field from your page, select it in the list on the right and click the  button. You can also double-click the field on the right to perform this same operation.
- The layout of this page and graphical organization of the fields on the page is, by default, automatically calculated by AssetCenter. You can, however, define the order of the fields' appearance on the page by ordering them in the left-hand pane using the buttons: , ,  and .

 Note:

You also have functions to control the geometry of the page (vertical and horizontal separators), available as keywords that you can add to the list of fields on the page. These functions are described later in this document.

7 Click **Modify**.

## Controlling the behavior of objects

When you add a database object to the page directly using the field above the **Fields and links** list, you can use control keywords, which enable you to:

- Define the type of the graphical control used to populate the field,
- Define the properties and behavior of the control.

The expected syntax is the following:

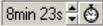
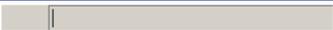
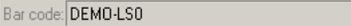
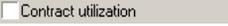
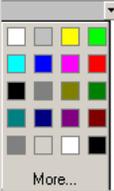
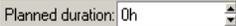
```
[<Control type>-]<SQL name of the object>[|<SQL name of a secondary object>] [|<Property of the control>="<Value>"*]
```

 Note:

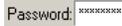
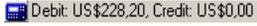
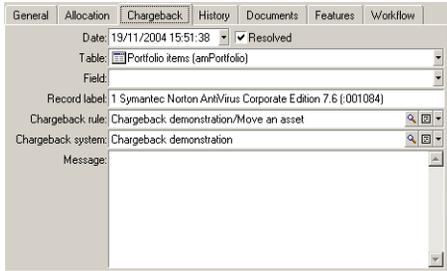
The SQL name of a secondary object is only used for certain properties.

## Types of graphic controls

The following table list and describes the available control types:

Control type	Description	Graphical example
TimerControl	Duration-entry control. By default, when displayed, the field is read-only and a time is started. To enter a duration in this type of control, click the icon opposite the field.	
Label	Text-data display control, read-only, without its label.	
CaptionedLabel	Text-data display control, read-only, with its label.	
MultilineLabel	Text-data display control, read-only, with its label, multi-line.	
Checkbox	Checkbox, with field label.	
ComboList	System itemized-list display control (closed, pre-defined list of values).	
Icon	Icon or image display control. The size is limited to the standard control height.	
ScrollablePicture	Image display control. Scrollbars are displayed if the image is too large.	No example
Picture	Image-display control.	
ColorCombo	Color-selection control.	
ComboDate	Date-display control.	
TimeSpanEdit	Duration-entry control.	
MultiLineEdit	Multi-line text-edit control.	

Control type	Description	Graphical example
Edit	Standard text data entry control.	
FileEdit	File-selection control.	
NumberEdit	Number-entry control.	
MonetaryEdit	Monetary field entry control.	
ItemizedList	Open itemized-list (customizable list of values) display control.	
LinkList	Linked records list display control.	
Notebook	Tabbed linked-records display control	
OneTo-ManyLinkEdit	Linked records display control, enabling list view or tabbed view depending on user choice.	
LinkEdit	Selection control for a record in a list.	
ReversePasswordEdit	Password-entry control. The encryption uses a reversible method thus enabling AssetCenter to use it to authenticate itself to external systems.	

Control type	Description	Graphical example
PasswordEdit	Password-entry control. This password is not reversible. Only AssetCenter is capable of authenticating passwords of this type.	
QueryCalc	Query-result display control. The icon representing a calculator, next to the control, enables you to recalculate the result of the query.	
DistantDet	Display control for a remote detail (i.e. outside of the table). This type of control enables you to display the detail of a record within another record.  <b>Note:</b>  The example shows the detail of a chargeback event displayed from an expense line.	
ExternalRef	Button-display control enabling you to view a record detail pointed to by a typed link (contextual link).	
TableChoiceList	Selection control for a table in the database.	
OneToOneLinkEdit	1-1 link selection control.	

## Graphical control properties

Property	Description	Control(s) using the property
UseLabel	If the property is set to 1: Display the field label opposite the control. If the property is set to 0: Do not display it.	All

Property	Description	Control(s) using the property
Caption	<p>The behavior of this property depends on the type of control:</p> <ul style="list-style-type: none"> <li>■ For OneToManyLinkEdit, LinkEdit, LinkList controls: The property defines the title of the selection window of the linked record.</li> <li>■ For Label, CaptionedLabel and MultilineLabel controls: The property displays a label whose value is that of a field in the table.</li> <li>■ For the QueryCalc control: The property defines the label explaining the calculated value.</li> </ul>	Label, CaptionedLabel, MultilineLabel, QueryCalc, LinkList, OneToManyLinkEdit, LinkEdit
Query	AQL query executed when the user clicks the  icon.	QueryCalc
ListConfig	<p>This property defines the columns in the list. The format used is the following:</p> <pre>ListConfig="&lt;SQL name of the object&gt;,&lt;Size of the column&gt;[,]*</pre> <p><b>Example:</b></p> <pre>ListConfig="Name,40,se Type,30"</pre>	LinkList, OneToManyLinkEdit

Property	Description	Control(s) using the property
WhereCond	<p>This property contains the Where clause of an AQL query. It enables you to filter the list of records in the drop-down list.</p> <p>For example, in the list of departments and employees, the <b>Supervisor</b> (Manager) link is associated with a control with the following WhereCond property:</p> <pre>WhereCond= "bDepartment=0"</pre> <p>Thus the departments will not appear in the list and will not be available for selection for the value of the supervisor.</p>	LinkList, LinkEdit
DistantDet	<ul style="list-style-type: none"> <li>■ SQL name of the detail displayed when viewing the detail of the linked record (🔗 icon) for the OneToManyLinkEdit and LinkEdit controls.</li> <li>■ SQL name of the distant detail displayed for the DistantDet control.</li> </ul> <p><b>Note:</b></p> <p>The filter of this detail is applied as well as the WhereCond property.</p>	OneToManyLinkEdit, DistantDet, LinkEdit

Property	Description	Control(s) using the property
AddMode	<p>Value 0 or 1. When adding a record to the list using the  button, this property enables you to define the default user-mode of the control:</p> <ul style="list-style-type: none"> <li>■ If the property is set to 1, the control is in creation mode and the user may create a record before selecting it to add it to the list.</li> <li>■ If the property is set to 0, the control is in selection mode only and the user must use the <b>New</b> button if they wish to create a record.</li> </ul>	OneToManyLinkEdit
OnZoom	This property contains the SQL name of an action to execute when the user clicks the  button.	OneToManyLinkEdit
OnAddLink	This property contains the SQL name of an action to execute when the user clicks the  button.	OneToManyLinkEdit
OnRemoveLink	This property contains the SQL name of an action to execute when the user clicks the  button.	OneToManyLinkEdit
Mode	<p>This property defines the default display mode for this type of control:</p> <ul style="list-style-type: none"> <li>■ If the property is set to 1, the control is displayed in list mode by default.</li> <li>■ If the property is set to 2, the control is displayed in tab mode by default.</li> </ul>	OneToManyLinkEdit

Property	Description	Control(s) using the property
DescString	<p>This property enables you force the description string of the window that presents the detail of the selected record (by clicking the  button).</p> <p>The format used by this property is exactly the same as that of a description string.</p> <p><b>Note:</b></p> <p>For further information on description strings, refer to the section <b>Table description strings</b> in the <b>Administration</b> guide.</p>	LinkEdit
ReadOnlyDetail	<p>This property enables you to define the display mode of the detail window of the selected window:</p> <ul style="list-style-type: none"> <li>■ If the property is set to 1, the detail is opened in read-only mode.</li> <li>■ If the property is set to 0, the open detail may be modified by the user.</li> </ul>	LinkEdit
AllowNone	<p>This property enables you to authorize or deny selection of the <b>No table</b> value in the selection list.</p> <ul style="list-style-type: none"> <li>■ If the property is set to 0, the value may not be selected.</li> <li>■ If the property is set to 1, the value may be selected.</li> </ul>	TableChoiceList

Property	Description	Control(s) using the property
AllowNoMainField	<p>This property enables you to display relationship (intermediary) tables (such as <b>amRelxxx</b>).</p> <ul style="list-style-type: none"> <li>■ If the property is set to 0, the relationship tables are not displayed.</li> <li>■ If the property is set to 1, the relationship tables are displayed.</li> </ul>	TableChoiceList
OnlyFeature	<p>This property enables you to display (or not display) the tables that have a corresponding table of feature values (such as <b>amFVxxx</b>).</p> <ul style="list-style-type: none"> <li>■ If the property is set to 0, all tables are displayed.</li> <li>■ If the property is set to 1, only the tables with a corresponding table of feature values are displayed.</li> </ul>	TableChoiceList
NoFeature	<p>This property enables you to display the feature value tables in the selection list.</p> <ul style="list-style-type: none"> <li>■ If the property is set to 0, all tables are displayed.</li> <li>■ If the property is set to 1, the feature value tables are not displayed.</li> </ul>	TableChoiceList

Property	Description	Control(s) using the property
HideSystem	<p>This property enables you to restrict the list of table to those included in the user license.</p> <ul style="list-style-type: none"> <li>■ If the property is set to 0, all tables are displayed.</li> <li>■ If the property is set to 1, only the tables authorized by the license are displayed.</li> </ul>	TableChoiceList

## Page layout

By default, the objects (fields or links) that you add to a page are displayed one after another in one single column. You can, however, define vertical and horizontal separators to modify this default behavior. These separators are defined by the following internal variables:

- **HSplitControl** for a horizontal separator.
- **VSplitControl** for a vertical separator.

To add a separator:

- 1 Open the **Contents** tab in the detail of a page.
- 2 Enter one of the two previously listed internal variables in the field above the list of fields and links.
- 3 Click .
- 4 The separator is transferred to the list of fields on the right. You can define the order in which these fields will appear on the page by using the , ,  and  buttons.

The following rules are applied when the page is displayed:

- All objects located between two **HSplitControl** variables will be part of the same horizontal block (one or more lines).
- The **VSplitControl** variable defines a separation in the column for the objects in a horizontal block.

For example, the contents of the General tab of an employee detail, as it is displayed by default in AssetCenter, can be defined as follows:

```
HSplitControl_0
HSplitControl_1_Adresse
HSplitControl_2
Location
```

```

HSplitControl_2
VSplitControl_2
Phone
MobilePhone
VSplitControl_2
Fax
HomePhone
HSplitControl_2
EMail
Field1
Field2
HSplitControl_1
HSplitControl_0
VSplitControl_0
IDNo
dHire
dLeave
VSplitControl_0
BarCode
Field3
HSplitControl_0
MultilineEdit-Comment

```

The underscore character enables you to define the relation of variables.

## Adding a page to a detail

To add a page to a detail:

- 1 Select the detail to which you want to add a page.
- 2 Select the **Pages** tab of the detail. You can now select the page(s) that you want to add to the detail. To do this:
  - Select in the list to the left (**Available pages**) the page(s) that you want to add to the detail, and click the  button to transfer them to the list on the right (**Pages in detail**). You can also double-click the pages on the left to transfer them to the right.
 

You can select a linked page, which will display a page linked to another table. To display a linked page, select its link from the drop-down list located above the **Available pages** list. This list presents all the 1-1 links that are available.
  - If you want to delete a page from your detail, select it in the list on the right and click the  button. You can also double-click the page on the left to perform this same operation.
  - To define the order of the tabs' appearance you need to order them in the right-hand pane using the buttons: , ,  and . The first page in

the list will be the first tab displayed; the second page will be the second tab appearing, etc.

## Propagating customizations to other supported languages

Objects added to the database are available in the language selected in AssetCenter Database Administrator during the modification.

The following objects must be created again for each additional language to be supported:

- Details
- Pages



Tip:

You must therefore successively select each language to be supported (list on the right side of the toolbar in AssetCenter Database Administrator) and create those objects in the same way as you did for the first language.

---

The following objects are created for all languages, but their labels and descriptions must be translated:

- Table
- Field
- Link
- Index

To translate the labels and descriptions of the objects added in a given language:

- 1 Select each additional language to be supported (the list is located on the right side of the toolbar in AssetCenter Database Administrator).
- 2 Select **Action/ Update descriptions**.
- 3 Translate the label and the description of each displayed object.
- 4 Click **Modify** to validate the changes.
- 5 Close the update window.

## Saving modifications

To save your modifications:

- 1 Select **File/ Save database structure** or click .
- 2 Follow the instructions given by the wizard.

## Verifying your modifications

To check your modifications, you need to launch AssetCenter and connect to the customized database. Using the method previously defined, we created a new table. To access this new table:

- 1 Select the **Administration/ List of screens** menu.
- 2 Search for your detail in the list that appears on screen and click **OK**. Your new detail will appear.
- 3 You can also create a view of this new table if you want: Select the **Tools/Views/Create view from current window** menu.
- 4 Enter a **Name** for your view. This name will appear in the Functions and favorites panel.
- 5 Select a functional **Domain**. The **Name** of your view will appear under this functional domain the Functions and favorites panel.
- 6 You can now directly access your new detail.

## Important note

In AssetCenter, all objects are defined by SQL names, in particular, the actions, view and screens. To open one of these objects from a menu or a function panel, AssetCenter uses the SQL name. If several objects, such as a view and a screen, have the same SQL name, AssetCenter tries to open the objects in the following order:

- View
- Screen

For example, let's say you create a detail for the Departments and Employees table. You name your detail amEmpDept. This is also the SQL name of the table, however. Now, when you select the **Portfolio/Departments and Employees** menu or click the **List of Employees** in the Functions and favorites panel, AssetCenter will open your detail and not the table.

---

## Customization prepared in a test database

When customizing an AssetCenter database, we recommend working on a copy of that database to preserve the integrity of the original.

This section details the different steps necessary for transferring the test database to the working database (of the same version of AssetCenter).

Here are the steps you must follow:

- 1 Make a copy of your working database. This will be your test database.

- 2 Develop and test the test database.

This test database contains the new data that will be categorized and transferred to the working database.

- 3 Propagate customizations from the test database to the production database

Instead of propagating the customizations from the test database to the production database, you can:

- 1 Export the additional data and the structure of the test database.

- 2 Import the additional data and the structure of the test database into the working database.

- 3 Migrate the data (the records) from the test database to the working database with Connect-It.

## Copying your working database

---

### Warning:

The way you copy your working database depends on your database engine. We recommend that you refer to the documentation provided with your database engine.

---

To copy a database to an empty database:

- 1 Create an empty shell with your SGBD.

- ▶ **Administration** guide, chapter **Creating, modifying and deleting an AssetCenter database**, section **Creating an empty shell with the DBMS**.

- 2 Create a DBMS login.

- ▶ **Administration** guide, chapter **Creating, modifying and deleting an AssetCenter database**, section **Creating an empty shell with the DBMS/Creating a DBMS login**.

- 3 Create a connection to the empty shell.

- ▶ **Administration** guide, chapter **Creating, modifying and deleting an AssetCenter database**, section **Creating a connection with AssetCenter**.

- 4 Start AssetCenter Database Administrator.

- 5 Connect to your working database (**File/Open/ Open existing database** menu).
- 6 Copy your working database ( **Action/Copy database to empty database** menu).

Select the connection to the empty shell that you have just created.

---



Note:

This operation copies the whole database (structure and data).

---

- ▶ **Administration** guide, chapter **Creating, modifying and deleting an AssetCenter database**, section **Changing your DBMS**.

## Propagating customizations from a test database to a production database

There are two ways to customize the production database:

- By modifying the **gbase\*. \* files** describing the structure of the database and importing those modifications into the production database.
- By modifying an empty test database and transferring those modifications to the production database.

Those customizations pertain to tables, fields, links, screens and indexes.

Propagating customizations to a production database is performed using AssetCenter Database Administrator

### Propagating customizations

To propagate customizations to a production database:

- 1 Connect to the production database.
- 2 Select **Action/ Integrate a customization**.
- 3 Follow the instructions given by the wizard.
- 4 Once the integration is over, check that the customizations done on the test database (or on the **gbase\*. \* files**) have been properly propagated to the production database.

### Logic of propagating customizations

The customizations made to your test database will overwrite the modifications made to your production database.

If you wish to keep the modifications made to your production database and integrate the customizations made to the test database, we recommend exporting the structures of both databases, compare them and integrate them using an integration program or a Connect-It scenario.

## Exporting the database structure

AssetCenter Database Administrator enables you to export the structure of your test database.

To export the structure of your test database:

- 1 Create a folder to hold the database structure.
- 2 Start AssetCenter Database Administrator.
- 3 Connect to your working database (**File/Open/ Open existing database** menu).
- 4 Export the structure of this database (**File/ Export the database structure** menu).

Exporting the structure of the database means exporting the following types of files:

- **.bin**
- **.cfg**
- **.xml**
- **.dsd**
- **.str**
- **.stt**
- **.usr**
- **.wiz**

## Importing the new database structure

---

### Warning:

We advise you to make a copy of your test database and your working database and save them. Importing the new structure is irreversible, and these copies will come in handy in case of any problem.

---

Importing the structure of your test database into your working database enables you to:

- Update your working database with the structure of the test database.
- Update the working database with the additional data from the test database.

To import the structure of your test database:

- 1 Start AssetCenter Database Administrator.
- 2 Connect to your working database (**File/Open/ Open existing database** menu).
- 3 Choose to update the structure (**Action/ Integrate a customization**).
- 4 Follow the instructions given by the wizard.

## Migrating data with Connect-It

Launch the Connect-It scenario corresponding to the data type to be migrated. Then include a WHERE clause to distinguish the modified data from the original data.

## Look out for software counters

When you create an entitlement counter with the **Create a counter** wizard, AssetCenter creates the queries that identify certain records by their primary keys.

Example of query associated with the **User rights scope** (LicQuery) of a counter:

```
(bDepartment = 0) AND (lEmplDeptId = 164)
```

If you create a counter in a test database, then transfer it to your production database, (export, then import), the identifiers that appear in the query most likely reference the wrong records.

For example, the identifier **164** could be **Mr. Doe** in the test database but **Mrs Dour** in the production database.

After importing the software counters and their queries in the production database, you must verify the queries associated with the counters.

 **Tip:**

Another solution consists of using more stable fields as the primary keys to identify the records in the queries: A name or a code, for example.

# Customization made by several operators at the same time

---

## Warning:

Implementing the process described in this section requires attention to detail and knowledge in the following areas:

- Structure of the AssetCenter database
- Exporting data
- Importing data
- Database administration
- Administering AssetCenter

Implementation may also be long and difficult if you wish to guarantee the integrity of the data in your production database.

---

## Explanation of the problem

This section explains how to customize the AssetCenter database as a team. This situation is restricting because the customizations that are saved in files cannot be shared.

As a consequence, each developer must:

- Work on a copy of the production database (called the **development database**).
- Save their own customizations separately in their own files.

You can then:

- 1 Merge the different customization files
- 2 Propagate the merged customizations to the production database

## Possible types of customizations

The AssetCenter database can be customized at different levels:

- Structural parameters of the database
- Data in the database

## Customizable structural parameters of the database

- Tables (label, description, etc.)
  - Fields (label, description, size, etc.)
  - Links (label, description, default value, etc.)
  - Indexes (label, description, list of fields, etc.)
  - Pages (label, description, content, etc.)
  - Screens (label, description, buttons, etc.)
  - Context-sensitive help for fields and links
- 

### Note:

These parameters sometimes use scripts (mandatory status of a field, for example).

---

## Customizable data in the database

- Features and feature classes
- Calculated fields
- Itemized lists
- Units
- Functional domains
- Actions
- Workflow schemes
- Scripts
- Queries
- Reports
- Forms
- Views
- User rights and profiles, access restrictions
- Functional rights
- Database options and tips of the day
- Configurations and preferences
- Workflows and deployment servers

---

 Note:

The choice of data to be customized depends on your needs.

Once you have made your choice, make sure the tables corresponding to this data cannot be modified by the AssetCenter users, exceptions permitted.

Make use of user profiles for this.

For example, you can stop tables linked to the workflow from being modified but still all itemized-list values to be modified.

You can be even more specific in defining the scope of configurations that can be made by using access restrictions.

If necessary, using prefixes in identification fields (such as **amWfScheme.Ref**) in the tables concerned, will enable you to easily identify the records that are concerned by the configuration.

In the case of itemized lists, using closed lists can provide a simple solution.

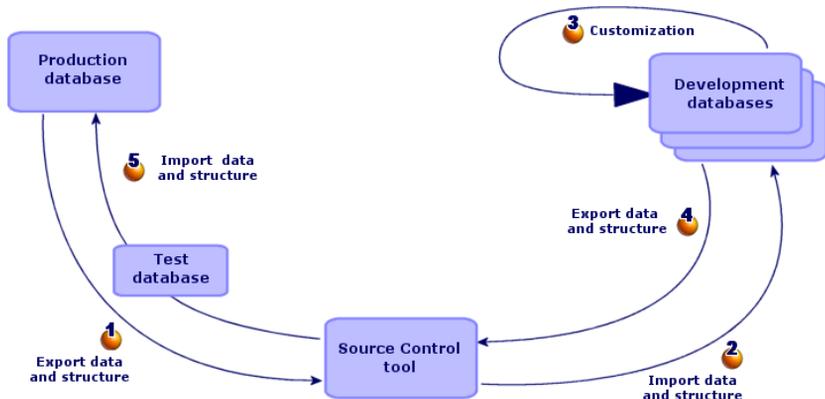
If you require a more flexible solution, you can define access restrictions for the itemized lists based on criteria of your own choice.

---

# Recommended methodology

## Global process

Figure 3.2. Customization by several operators - simplified process



To customize the AssetCenter as a team:

- 1 Exporting data and structure of the production database [page 72]
  - 1 Exporting the structure of the production database [page 73]
  - 2 Exporting the data to be customized from the production database [page 74]
  - 3 Storing the result of the export in a source control tool [page 77]
  - 4 Creating test data [page 78]

Note:

This series of steps is only performed once at the initial implementation stage of the project.

- 2 Importing the data and structure of development databases [page 78]
  - 1 Synchronizing files from the source control on the development computers [page 78]

- 2 Check out the files from the source control [page 78]
  - 3 Creating development databases and importing data to customize and test data [page 79]
- 

 Note:

This series of steps if performed once at the start and then whenever required afterwards.

---

- 3  Customizing the development databases [page 83]
  - 1 Customizing the structure of the development databases [page 83]
  - 2 Customizing the data in the development databases [page 84]
- 4  Exporting the data and structure of development databases [page 84]
  - 1 Exporting the data of the customized development databases [page 84]
  - 2 Exporting the customized data from the development databases [page 85]
  - 3 Updating the files in the source control tool [page 85]
- 5  Importing data and structure into the production database [page 85]
  - 1 Creating a test database and importing the merged customized data [page 85]
  - 2 Testing the test database [page 86]
  - 3 Applying the customized structure to the production database [page 86]
  - 4 Importing the customized data in the production database [page 87]

## Exporting data and structure of the production database

Because you are not going to directly modify the production database, you must export the data and structure of the database in the form of text files.

The text format enables you to manage the resulting files in a revision control (or source control) tool.

The files obtained during the export enable you to:

- Keep a trace of the structure of the production database prior to customization
- Create development databases to make and test customizations
- Compare the files generated after customizing the development databases with the original files

## Exporting the structure of the production database

To export the structure of the production database:

- 1 Create a folder in which the results of the export will be stored
- 2 Start AssetCenter Database Administrator
- 3 Connect to the production database (menu **File/ Open/ Open existing database**)
- 4 Export the structure of this database (**File/ Export the database structure** menu).

Exporting the structure of the database creates the following files:

- **<Connection name>.bin**: Definition of casual users (not modifiable)
- **1stpage\_xx.cfg**: Description of the pages
- **1stdet\_xx.cfg**: Description of the screens
- **<Connection name>.xml**: Structure of the database
- **<Connection name>\_xx.xml**: Database dictionary
- **dawkfl.dsd**: Definition of the automation workflow (not modifiable)
- **tips.str**: Tips of the day
- **<Connection name>\_xx.str**: Contextual help on fields and links
- **<Connection name>.wiz**: Functions library for wizards

---

 **Note:**

**xx** corresponds to the language version of **AssetCenter** (depending on availability):

- **en**: English
  - **fr**: French
  - **de**: German
  - **it**: Italian
  - **sp**: Spanish
  - **ja**: Japanese
-

## Exporting the data to be customized from the production database

---

### Note:

If your production database does not yet contain data, this section is still pertinent as for the creation of import scripts, which will be used throughout the process.

---

### To export the data to be customized:

- 1 Create a folder to hold the data from the database.
  - 2 Start AssetCenter Export
    - ▶ **Advanced use guide**, chapter **Exporting data and creating SQL views**
  - 3 Connect to the production database
  - 4 Select the **Edit/ Options** menu.
  - 5 Unfold the branch **Edit**
  - 6 Set the **Displaying of database objects** to **SQL name (Description)**
  - 7 Confirm this setting (**OK** button)
  - 8 Display a new export script (**File/ New script** menu)
- 

### Tip:

You can also create your export script using the **custom.scx** template provided with the sample data.

This script is located in a sub-folder of **C:\Program Files\Peregrine\AssetCenter\datakit\custom**.

▶ The **custom.scx** script respects the indications in section [Database customization - Tables to take into consideration](#) [page 87].

---

- 9 Save this script under the name of your choice (**File/ Save script** menu).
- 

### Important:

Save regularly any modifications you make to the script.

---

- 10 Add a query for each data set to be exported - In general, one data set per table (**New** button)  
List of tables to take into consideration: ▶ [Database customization - Tables to take into consideration](#) [page 87]
- 11 For each query:
  - 1 Name the query (**Name** field)



Tip:

We recommend using the SQL name of the exported table as the name of the query.

Example: **amAction**

---

- 2 Define the full path of the file that will be used to stored the exported data (**File** field)
- 



Tip:

We recommend using the SQL name of the exported table as the file name.

Example: **amAction.txt**

---

- 3 Click the magnifier next to the **Query** field.
- 4 Add the following elements from the list on the left to the list on the right:
  - All fields, except:
    - **Modified on** (dtLastModif)
    - **Hierarchic level** (sLv)
 They are identified by the  icon.
  - All Memo fields.
 

They can be accessed from the list of links.

 They are identified by the  icon.
  - The useful 1 links (no n link).
 

They are identified by the  or  icon.

 For these links, select the field or fields that constitute a relevant reconciliation key.
    - ▶ [Database customization - Tables to take into consideration](#) [page 87]
 Do not use a primary key.
- 5 Enable sorting on the fields making up the reconciliation key (red arrow with a digit).

 **Warning:**

This is required so that the exported files can be management more easily with a source control tool.

If the files are not adequately sorted, the source control application will not be able to analyse the differences between the files to be reconciled.

 **Important:**

Exception: If there is a **Full name** field (FullName) in the table, enable the sort on the **Full name** only, and clear the **Visibility** check box.

By enabling the sort on the **Full name** field, you can be sure of importing the parent records before the child records. In this way, you are sure of being able to populate the parent link.

By clearing the **Visibility** checkbox, the **Full name** field is not exported. You risk not being able to import it later on.

During the import, this field is automatically generated from the parent/child links.

- 6 Confirm your choices by clicking **OK**
- 7 In the **Query** field, after each of the fields making up the reconciliation key of the table, add **keyvalue**.

Example:

```
SELECT sX, sY, seType, Comment.memComment, WfTransition.Ref keyvalue
, WfScheme.Ref keyvalue FROM amWfSysActiv ORDER BY lSysActivId
```

In this example, we have added **keyvalue** after the **WfTransition.Ref** and **WfScheme.Ref** fields because they make up the reconciliation key for records in the **amWfSysActiv** table.

 **Note:**

AssetCenter Export uses the values of the **keyvalue** fields for certain files created during the export.

- 8 In the **Query** field, if there are links to images (**Images** table (amImage)), replace the image identifier with **<link>.Name**.

Example:

```
SELECT Name, SQLName, Comment.memComment, WfActivity.Ref keyvalue, WfEvent.Ref, lIconId FROM amWfUserChoice ORDER BY lUserChoiceId
```

Becomes:

```
SELECT Name, SQLName, Comment.memComment, WfActivity.Ref keyvalue, WfEvent.Ref, Icon.Name FROM amWfUserChoice ORDER BY lUserChoiceId
```

In this example, **llconId** has been replaced by **Icon.Name**.

- 12 Display the **Formatting** tab
- 13 Populate the **Column title** with the value **SQL name**
- 14 Save the script (**File/ Save script** menu)
- 15 Execute the script (**Actions/ Execute the script**)
- 16 View the result of the export: You obtain 1 file per query + 1 file per memo field and per record exported.  
Make sure that the result of the export is consistent.

### Storing the result of the export in a source control tool

This enables you to:

- Keep a trace of the database structure and customized data each time a change is made.  
You can thus return to a previous status if necessary and make comparisons of different versions.
- Merging files modified by different persons.

---

 **Note:**

Merging different file versions is part of the base functionality of a source control tool.

---

Add the files obtained during the export processes to your source control tool.

 **Important:**

From now on, you must use these files as a starting point when making any modifications to the database structure or when customizing the database.

### To store the result of the export in a source control tool:

Refer to the documentation of your source control tool.

## Creating test data

When making customizations, sometimes you need to use test data for working with development database but which is not useful to be stored in the source control.

For example, when creating an action, an author needs to be associated to it. The author is selected from the **Employees** table (amEmplDept). It is therefore necessary to have employee records in the development database. However, these employee records do not need to be customized. For this reason, the employees will not be stored in the source control.

We recommend that you create a test data set starting out from:

- Data from the production database:
  - Records in the target tables (**Employees**, for example) and the links (**Author**, for example) in the tables to be customized (**Actions**, for example).
  - All or a selection of record in the tables required for tests.
- Other data, if necessary.

## Importing the data and structure of development databases

### Synchronizing files from the source control on the development computers

Before customizing a database on their computers, the developers must create a database whose data is synchronized with the data on the source control.

### To synchronize files from the source control on the development computers:

- 1 Install and configure the source control tool
- 2 Synchronize the database structure files on the development computer

### Check out the files from the source control

Each developer must identifier what they are going to modify and check out the corresponding files.

### To check out the files from the source control:

Refer to the documentation of your source control tool.

## Creating development databases and importing data to customize and test data

In order that each developer be able to customize the database independently, create a development database for each developer.

### To create a development database:

- 1 Create an empty shell with your SGBD
  - ▶ **Administration** guide, chapter **Creating, modifying and deleting an AssetCenter database**, section **Creating an empty shell with the DBMS**.
- 2 Create a DBMS login
  - ▶ **Administration** guide, chapter **Creating, modifying and deleting an AssetCenter database**, section **Creating an empty shell with the DBMS/ Creating a DBMS login**.
- 3 Create a connection to the empty shell
  - ▶ **Administration** guide, chapter **Creating, modifying and deleting an AssetCenter database**, section **Creating a connection with AssetCenter**.
- 4 Start AssetCenter Database Administrator
- 5 Open the **<Connection name>.xml** file synchronized from the source control on the development computer (**File/ Open** menu, **Open database description file - create new database** option).  
This file is obtained in the step [Exporting the structure of the production database](#) [page 73].
- 6 Start the database creation wizard (**Action/ Create database** menu).
- 7 Populate the pages of the wizard as follows (navigate through the wizard pages using the **Next** and **Previous** buttons):  
**Generate SQL script / Create database** page:

Fields	Value
Database	Select the connection to the empty shell created with AssetCenter.
Creation	Create the database
Use advanced creation options	Select this option if you need to define the tablespaces for tables, indexes and LOBs.

**Creation options** page:

 Note:

The tablespaces must be created beforehand using the DBMS.

**Creation parameters page:**

Fields	Value
Password	<p>Password to associate with the administrator login.</p> <p>Note:</p> <p>The AssetCenter database administrator is the record in the <b>Departments and employees</b> (amEmplDept) table for which the <b>Name</b> (Name) field is set to <b>Admin</b>.</p> <p>The database connection login is stored in the <b>User name</b> (UserLogin) field. The administration name is <b>Admin</b>.</p>
Confirm password	Confirm the value entered in the <b>Password</b> field.
License file	Select the license file <b>License.cfg</b> provided with AssetCenter.

**Create system data page:**

Fields	Value
Use time zones	<p>Select this option in order to be able to manage the time-zone differences between the database server, the date field and the AssetCenter clients.</p> <p>► <b>Advanced use guide</b>, chapter <b>Time zones</b>.</p>
Server	<p>Select the time zone of the AssetCenter database server.</p> <p>This is the time zone defined, for example, at the level of the Windows Control Panel of the server.</p>
Data	Select the time zone in which the dates stored in the database will be expressed.
Use help on fields	Select this option to insert the information describing the database fields and links.

Fields	Value
Use AutoCAD integration	Select this option if you plan on using the AutoCAD integration functionality in AssetCenter. ► <b>Administration</b> guide, chapter <b>AutoCAD integration</b> .

### Data to import page:

Fields	Value
Available data	Do not select any data.
Stop import if error	Do not select this option
Log file	Do not populate this field.

- Execute the options defined using the wizard (**Finish** button).

### To import the data to be customized in a development database:

- Start AssetCenter
- Connect to the development database
- Start the import function (**File/ Import, Import database** button, **Text** tab, **Open** button)
- Select the **Edit/ Options** menu.
- Clear the option **Do not import empty fields from the source database**
- Confirm this setting (**OK** button)
- Save the blank import script using the name of your choice (**File/ Save** menu).

#### Important:

Save regularly any modifications you make to the script.

#### Tip:

You can also create your export script using the **custom.scx** template provided with the sample data (**File/ Open script**)

This script is located in a sub-folder of **C:\Program Files\Peregrine\AssetCenter\datakit\custom**.

► The **custom.scr** script respects the indications in section [Database customization - Tables to take into consideration](#) [page 87].

- Configure the script so that it imports the exported data files

- ▶ **Administration** guide, chapter **Import**.
- 9 For each file to be imported (1 for each file exported; ▶ [Database customization - Tables to take into consideration](#) [page 87]):
  - 1 Add the file (**File/ Add**)
  - 2 Populate the pages displayed by the wizard, in particular:
    - Page 1: Select the options **First imported line contains column names** and **Separators**
    - Page 2: Select the option **Tab**
    - Page 3: Select the columns containing **Date** or **Date + Time** data and populate the **Type** field accordingly
  - 3 Select the target table: Main table matching the imported file
  - 4 Map the file to the table (**Edit/ Map**)
  - 5 Select the mapping (**file, target table**)
  - 6 Map the columns in the file to the fields in the target table (**Edit/ Properties** menu).



**Tip:**

The best is to use the  button to automatically map the fields based on their SQL names.

---

For certain fields, a **File**-type calculation formula must be created and mapped to the target field instead of the field.

This is case of the following fields in particular:

- Memo fields
- **Data** fields (blbData)
- **Script** fields (memScript)
- **Definition** field in the **Functional rights** table
- **Period** field in the **Workflow events** table (amWfEvent)

When you use the  button, the formulas are automatically created and associated with the correct field.

If you look at details of one of these formulas, you will see:

- **Type of field** field: **File**
- **Main field** field: Column name of the exported file indicating the file name storing the memo

For duration fields (their SQL names start with **ts** for timespan), you must also create a formula with the following features:

- **Field type: Script**
- **Script: RetVal = [<SQL name of the field>] & "s"**  
Example: **RetVal = [tsDelay] & "s"**

Associate the calculate field you have just created with the target field (instead of the source field associated with it by default).

- 7 Define the keys recommended in the table [Database customization - Tables to take into consideration](#) [page 87] as the reconciliation keys for the main table and linked tables

---

 **Warning:**

The  button does not automatically select the appropriate keys. You must systematically verify them.

- 
- 10 Save the script (**File/ Save** menu)
  - 11 Verify the import settings (**Edit/ Option** menu)
  - 12 Execute the script (**Importer** button)
  - 13 Make sure that the result of the import is consistent.

---

 **Important:**

These instructions should suffice for all tables except for the workflow schemes, for which you will have to test the scripts in-depth to get them to work correctly.

---

## To import the test data in a development database:

Create a new import script in the same way as for importing the data to customize.

## Customizing the development databases

### Customizing the structure of the development databases

The structure of the database may be modified by more than one developer at the same time.

For each individual developer, to customize the structure of the development database:

- 1 Start AssetCenter Database Administrator

- 2 Connect to your instance of the development database (**File/Open/Open existing database** menu).
- 3 Modify the structure of the database
- 4 Save the modifications (**File/ Save** menu)

## Customizing the data in the development databases

The database data may be modified by more than one developer at the save time.

### To customize the data in a development database:

- 1 Start AssetCenter
- 2 Connect to the development database (menu **File/ Open/ Open existing database**)
- 3 Modify the data in the development database



Tip:

Populate the key recommended in the table [Database customization - Tables to take into consideration](#) [page 87] so that it enables you to identify the developer who created or modified the record.

You can, for example, start each identifier with a prefix for each developer.

---



Tip:

When performing your tests, you may have to create temporary records.

Don't forget to delete these temporary records.

If the number of these records ever becomes too great, you may wish to consider creating separate test and development databases.

---

## Exporting the data and structure of development databases

### Exporting the data of the customized development databases

This operation must be performed for each development database that has been customized.

### To export the structure of a development database:

- 1 Start AssetCenter Database Administrator

- 2 Connect to the development database (menu **File/ Open/ Open existing database**)
- 3 Export the database structure (**File/ Export the database structure**) to the place where you synchronized the structure files

## Exporting the customized data from the development databases

To export the data of a development database:

- 1 Start AssetCenter Export
- 2 Connect to your instance of the development database
- 3 For each script corresponding to data to be exported:
  - 1 Open the script (**File/ Open script**)
  - 2 Execute the script (**Actions/ Execute the script**)
    - ▶ [Exporting the data to be customized from the production database \[page 74\]](#)

## Updating the files in the source control tool

When a developer finishes making their modifications, they check the exported files back into the source control.

If during the time they have been working on a file, the centrally stored version of the file in the source control has been updated by another user, they will have to merge the modifications before checking the file back in, so as not to overwrite the changes made by the other users.

The source control tool can be used in conjunction with a **diff and merge** tool, which highlights the modifications made between successive versions.

## 5 Importing data and structure into the production database

### Creating a test database and importing the merged customized data

When every developer has customized their development database, and the results have been placed back in the source control, you must verify the application of the new structure and importing the customized data in a test database.

## To create a test database with an up-to-date structure:

Proceed in the same way as in the step [Creating development databases and importing data to customize and test data](#) [page 79].

## To import the customized data into the test database:

Proceed in the same way as in the step [Creating development databases and importing data to customize and test data](#) [page 79].

## To import the test data into the test database:

Proceed in the same way as in the step [Creating development databases and importing data to customize and test data](#) [page 79].

## Testing the test database

Make sure the structure of the database matches the customizations made in the development databases.

Also test one or more customized objects in the development databases.

In the case of actions, for example, execute one or two actions to check whether they function correctly.

## Applying the customized structure to the production database

If the tests on the test database are successful, you can update the production database.

## To apply the customized structure to the production database:

- 1 Start the source control tool
- 2 Synchronize the folder containing the structure files and data
- 3 Start AssetCenter Database Administrator
- 4 Open the production database
- 5 Select **Action/ Integrate a customization**
- 6 Select the **Integrate from a database description file** option
- 7 **Working folder:** Temporary folder of your choice.
- 8 **Source file:** Full path of the structure file, **<Database name> .xml**, that you have synchronized (do not select the file, **<Database name>\_xx.xml**)
- 9 Click **Integrate**

- 10 Once the integration is performed, make sure the customizations have been correctly made in your production database

## Importing the customized data in the production database

To import the customized data into the test database:

Proceed in the same way as in the step [Creating development databases and importing data to customize and test data](#) [page 79].

## References: Customizable tables

The following is a list of customizable objects.

For each of these objects, the related tables are indicated.

For each of these tables, a recommended reconciliation key is given as are the useful 1 links (Memo links and Images links are not given).

### Tip:

During an import, the order of the tables is important because certain imported records reference records in other tables that must already exist. The order of the tables listed below must respect the dependencies between tables.

Table 3.3. Database customization - Tables to take into consideration

Table	Recommended key	Useful 1 links
<b>Images</b>		
Images (amImage)	Name (Name)	
<b>Calculated fields</b>		
Calculated fields (amCalcField)	SQL name (SQLName) + Table (TableName)	
<b>Scripts</b>		
Scripts (amScriptLibrary)	SQL name (SQLName)	
<b>Units</b>		
Units (amUnit)	Symbol (Symbol)	
<b>Deployment</b>		
Deployment workflows (am-DaWfScheme)	Reference (Ref)	

Table	Recommended key	Useful 1 links
Deployment servers (amDaServer)	Name (Name)	Computer.Asset tag (Computer.AssetTag)
<b>Functional domains</b>		
Functional domains (amFuncDomain)	SQL name (SQLName)	Parent domain (Parent.Code)
<b>Queries</b>		
Queries (amQuery)	SQL name (SQLName)	Author.IDNo (Author.IDNo) Domain.SQL name (Domain.SQLName)
<b>Views</b>		
View (amViewDef)	SQL name (SQLName)	Author.IDNo (Author.IDNo) Domain.SQL name (Domain.SQLName)
<b>Options of the application</b>		
Applications options (amOption)	<ul style="list-style-type: none"> <li>■ Section (OptSection)</li> <li>■ Entry (OptEntry)</li> </ul>	Employee.ID # (EmpDept.IDNo)
<b>Configurations and preferences</b>		
Configurations and preferences (amSysConfig)	<ul style="list-style-type: none"> <li>■ Entry (CfgEntry)</li> <li>■ Section (CfgSection)</li> <li>■ Employee (EmpDept.IDNo)</li> </ul>	Employee.ID # (EmpDept.IDNo)
<b>Reports</b>		
Reports (amReport)	SQL name (SQLName)	Author.IDNo (Author.IDNo) Domain.SQL name (Domain.SQLName)
<b>Forms</b>		
Internal forms (amForm)	SQL name (SQLName)	Author.IDNo (Author.IDNo) Domain.SQL name (Domain.SQLName)
<b>Itemized lists</b>		
Itemized lists (amItemizedList)	Identifier (Identifier)	
Itemized list value (amItemListVal)	<ul style="list-style-type: none"> <li>■ Value (Value)</li> <li>■ Itemized list.Identifier (ItemizedList.Identifier)</li> </ul>	Itemized list.Identifier (ItemizedList.Identifier)
<b>Features</b>		
Feature classes (amFeatureClass)	Code (Code)	Parent class (Parent.Code)
Features (amFeature)	SQL name (SQLName)	Class.Code (FeatureClass.Code) Itemized list.Identifier (ItemizedList.Identifier) Unit.Symbol (Unit.Symbol)

Table	Recommended key	Useful 1 links
Features (amFeature)	Export the records a second time with the <b>SQL name (SQLName)</b> and <b>Source features.SQL name (SrcFeatures.SQLName)</b> only	
	<b>Note:</b>	
	This second import enables you to create the links between consolidating features and consolidated features during the import phase.	
	This can only be achieved if all these features already exist.	
	This condition is ensured by importing the first file.	
	AssetCenter Export generates one line per link.	
Feature parameters (amFeat-Param)	<ul style="list-style-type: none"> <li>■ Table (TableName)</li> <li>■ Feature.SQL name (Feature.SQLName)</li> </ul>	<a href="#">Help on this parameter.Description (ShortHelp.Comment)</a> <a href="#">Help on this parameter.Example (ShortHelp.Example)</a> <a href="#">Help on this parameter.Precautions (ShortHelp.Warning)</a> <a href="#">Feature.SQL name (Feature.SQLName)</a>
<b>Actions</b>		
Actions (amAction)	SQL name (SQLName)	<a href="#">Author.IDNo (Author.IDNo)</a> <a href="#">Deployment server.Name (DaServer.Name)</a> <a href="#">Deployment workflow.Reference (DaWfScheme.Ref)</a> <a href="#">Domain.SQL name (Domain.SQLName)</a> <a href="#">Action to execute.SQL name (ExecAction.SQLName)</a> <a href="#">Form.SQL name (Form.SQLName)</a> <a href="#">Report.SQL name (Report.SQLName)</a>
<b>User profiles</b>		
User profiles (amProfile)	SQL name (SQLName)	<a href="#">Default action.SQL name (DefLoginAction.SQLName)</a>
User rights (amUserRight)	SQL name (SQLName)	
Access restrictions (amAccess- Restr)	SQL name (SQLName)	
Functional rights (amFuncRight)	SQL name (SQLName)	

Table	Recommended key	Useful 1 links
User rights (amUserRight)	Export the records a second time with the <b>SQL name</b> (SQLName) and <b>Profiles.SQL name</b> (Profiles.SQLName) only	
	<b>Note:</b>	
	This second import enables you to create the links between user profiles and user rights during the import phase.	
	AssetCenter Export generates one line per link.	
Access restrictions (amAccess- Restr)	Export the records a second time with the <b>SQL name</b> (SQLName) and <b>Profiles.SQL name</b> (Profiles.SQLName) only	
	<b>Note:</b>	
	This second import enables you to create the links between user profiles and access restrictions during the import phase.	
	AssetCenter Export generates one line per link.	
Functional rights (amFuncRight)	Export the records a second time with the <b>SQL name</b> (SQLName) and <b>Profiles.SQL name</b> (Profiles.SQLName) only	
	<b>Note:</b>	
	This second import enables you to create the links between user profiles and functional rights during the import phase.	
	AssetCenter Export generates one line per link.	
<b>Workflow schemes</b>		
Employee groups (amEmpGroup)	Barcode (BarCode)	Supervisor.IDNo (Supervisor.IDNo) Sub-group of.Barcode (Parent.BarCode)
Employee groups (amEmpGroup)	Export the records a second time with the <b>Barcode</b> (BarCode) and <b>Members.IDNo</b> (Members.IDNo) only	
	<b>Note:</b>	
	This second import enables you to create the links between workflow groups and members during the import phase.	
	AssetCenter Export exports one line per line between a group and a member.	

Table	Recommended key	Useful 1 links
Workflow roles (amWfOrgRole)	Reference (Ref)	Assignee.ID No (Assignee.IDNo) Group.Barcode (EmplGroup.BarCode)
Workflow schemes (amWfScheme)	Reference (Ref)	Supervisor.ID # (Supervisor)
System activities (amWfSysActiv)	<ul style="list-style-type: none"> <li>■ Workflow.Reference (WfScheme.Ref)</li> <li>■ Type (seType)</li> </ul>	Workflow.Reference (WfScheme.Ref)
<p><b>Note:</b></p> <p>In this first file, add a WHERE clause to filter the START activities (seType = 0).</p>		
Workflow activities (amWfActivity)	<ul style="list-style-type: none"> <li>■ Workflow.Reference (WfScheme.Ref)</li> <li>■ Reference (Ref)</li> </ul>	Assignee.Reference (Assignee.Ref) Calendar.SQL name (Calendar.SQLName) Action.SQL name (NotifAssignAct.SQLName) Template.Reference (Template.Ref) Template.Workflow.Reference (Template.WfScheme.Ref) Workflow.Reference (WfScheme.Ref) Wizard.SQL name (Wizard.SQLName)
User choices (amWfUserChoice)	<ul style="list-style-type: none"> <li>■ SQL name (SQLName)</li> <li>■ Activity.Reference (WfActivity.Ref)</li> <li>■ Activity.Workflow.Reference (WfActivity.WfScheme.Ref)</li> </ul>	Activity.Reference (WfActivity.Ref) Activity.Workflow.Reference (WfActivity.WfScheme.Ref)
Workflow events (amWfEvent)	<ul style="list-style-type: none"> <li>■ Workflow.Reference (WfScheme.Ref)</li> <li>■ Source activity.Reference (SrcActivity.Ref)</li> <li>■ Reference (Ref)</li> </ul>	Source activity.Reference (SrcActivity.Ref) Workflow.Reference (WfScheme.Ref)

Table	Recommended key	Useful 1 links
Workflow transitions (amWfTransition)	<ul style="list-style-type: none"> <li>■ Reference (Ref)</li> <li>■ Source event.Reference (SrcEvent.Ref)</li> <li>■ Source event.Source activity.Reference (SrcEvent.SrcActivity.Ref)</li> <li>■ Workflow.Reference (WfScheme.Ref)</li> </ul>	<a href="#">Source event.Reference (SrcEvent.Ref)</a> <a href="#">Source event.Source activity.Reference (SrcEvent.SrcActivity.Ref)</a> <a href="#">Target activity.Reference (TargetActiv.Ref)</a> <a href="#">Workflow.Reference (WfScheme.Ref)</a>
System activities (amWfSysActiv)	<ul style="list-style-type: none"> <li>■ Workflow.Reference (WfScheme.Ref)</li> <li>■ Transition.Reference (WfTransition.Ref)</li> <li>■ Transition.Source event.Reference (WfTransition.SrcEvent.Ref)</li> <li>■ Transition.Source event.Source activity.Reference (WfTransition.SrcEvent.SrcActivity.Ref)</li> </ul>	<a href="#">Transition.Reference (WfTransition.Ref)</a> <a href="#">Transition.Source event.Reference (WfTransition.SrcEvent.Ref)</a> <a href="#">Transition.Source event.Source activity.Reference (WfTransition.SrcEvent.SrcActivity.Ref)</a> <a href="#">Workflow.Reference (WfScheme.Ref)</a>

**Note:**

In this second file, add a WHERE clause to filter the END activities (seType = 1).

Workflow activities (amWfActivity)	Export the records a second time with the <b>Workflow.Reference</b> (WfScheme.Ref), <b>Reference</b> (Ref) and <b>Actions.SQL name</b> (Actions.SQL name) only
------------------------------------	--

**Note:**

This second import enables you to create the links between workflow groups and actions during the import phase.

AssetCenter Export generates one line per link.

Workflow alarms (amWfActivAlarm)	<ul style="list-style-type: none"> <li>■ Reference (Ref)</li> <li>■ Activity.Reference (WfActivity.Ref)</li> <li>■ Activity.Workflow.Reference (WfActivity.WfScheme.Ref)</li> </ul>	<a href="#">Action.SQL name (Action.SQL-Name)</a> <a href="#">Activity.Reference (WfActivity.Ref)</a> <a href="#">Activity.Workflow.Reference (WfActivity.WfScheme.Ref)</a>
----------------------------------	---	---

**Calendars**

Time zones (amTimeZone)	Code (Code)
-------------------------	-------------

Table	Recommended key	Useful 1 links
Calendar of business days (amWorkCalendar)	SQL name (SQLName)	Time zone.Code (TimeZone.Code)
<b>Currencies</b>		
Currencies (amCurrency)	Name (Name)	
Exchange rates (amCurRate)	<ul style="list-style-type: none"> <li>■ Source currency.Name (SrcCurrency.Name)</li> <li>■ Target currency.Name (TargetCurrency.Name)</li> </ul> Date of rate (dtStart)	<ul style="list-style-type: none"> <li>Source currency.Name (SrcCurrency.Name)</li> <li>Target currency.Name (TargetCurrency.Name)</li> </ul>

## Remarks: Choice of reconciliation keys

The choice of reconciliations keys is tricky:

- You may not use a primary key as a reconciliation key. In effect, each developer will be generating primary keys that are specific to their own development databases. However, these identifiers will be different once they are imported into the production database (you cannot import a primary identifier; AssetCenter generates them automatically during the import process).
- You require a reconciliation key that will enable you to update the records in the production database whenever it is customized again in the future.

Example:

- 1 During an initial round of customizations, a developer creates a new action, **A1**, in a development database.  
The primary key of the action is **PK1** and its SQL name **SQL1**.
- 2 When it is imported into the production database, the primary key of the action **A1** automatically changes (**PK2**). However, its SQL name remains the same (**SQL1**).
- 3 During a subsequent round of customizations, a developer modifies the action **A1** in a development database.  
The primary key of the action is **PK1** and its SQL name **SQL1**.
- 4 When it is imported into the production database, the action **A1** is identified in the production database by its SQL name (**SQL1**), which is used as a reconciliation key. The record describing the action **A1** is updated.



# 4 Database options

## CHAPTER

Certain options specifically concern a database.

They are common to all client computers connecting to the database.

This chapter explains the options in question and how to configure them.

---

## To configure the database options

1 Launch AssetCenter.

2

 **Warning:**

Only the **Admin** user and users for which the **Administration rights** option has been selected are able to configure the database options.

---

Connect to the database (**File/Connect to database** menu).

3 Display the list of database options (**Administration/Database options** menu).

4 Select the option to modify.

5 Proceed using one of the following methods:

- Double click the value of the option in the **Current value** column.
- Press **Shift + Space bar**.

6 The value can be modified in different ways:

- If the value is part of a list: Select the value from the list.
  - If the value is entered freely and is on one single line: Enter the value in the **Current value** column.
  - If the value is entered freely and is over several lines: enter the value in the edit zone at the bottom of the option window.
- 

 **Tip:**

If this window is not shown, place the mouse pointer over the lower edge of the options window until the split pointer appears, then drag it upwards.

---

- 7 Validate the options (**OK** button).
- 8 Have the users reconnect to the database in order for the new options to be taken into account at the level of their AssetCenter client.

---

## Database option configuration window

---

 **Note:**

Values displayed in black cannot be modified.

Values displayed in black can be modified.

The **Default value** column indicates the standard values for reference.

---

Table 4.1. Database options - Description of the available options

Section	Name of the option	Description	Example of values
Procurement	Let AssetCenter Server create the items received in the portfolio	<ul style="list-style-type: none"> <li data-bbox="732 291 1160 812"> <p>■ <b>Yes:</b> Receiving an item from a purchase order will create a record in the <b>Receiving slips</b> table (amReceipt).</p> <p>However, the received items are not created in their respective tables (assets, computers, portfolio items, etc.).</p> <p>Temporary records are created in the <b>Items received</b> table (amItems-Received) and processed by the <b>Create assets, consumables, etc. corresponding to items received</b> module in AssetCenter Server, which takes care of creating records in the final tables.</p> <p>Purpose of this method: Giving this task to AssetCenter Server rather AssetCenter enables increased performance for those client computers used to receive purchase orders.</p> </li> <li data-bbox="732 994 1160 1251"> <p>■ <b>No:</b> Receiving an item from a purchase order will create a record in the <b>Receiving slips</b> table (amReceipt).</p> <p>The corresponding records are immediately created in their respective tables (assets, computers, portfolio items, etc.).</p> </li> </ul>	Yes

Section	Name of the option	Description	Example of values
Procurement	Create expense lines for consumables as soon as received	<ul style="list-style-type: none"> <li>■ <b>Yes:</b> As soon as a consumable is received, an expense line is generated. It corresponds to the purchase of the consumable. An additional expense line is not created when the consumable is used (i.e. when the consumable is linked to the parent asset that consumes it). You must create a chargeback rule if you want to generate expense lines taking into account the <b>consumption</b> aspect.</li> <li>■ <b>No:</b> No expense line is created when a consumable is created upon its receipt. An expense line is created when the consumable is used (i.e. when the consumable is linked to the parent asset that consumes it).</li> </ul>	Yes
Import	Disable history during import	<ul style="list-style-type: none"> <li>■ <b>Yes:</b> No record is created in the <b>History</b> table (amHistory) for modifications brought about by importing the data.</li> <li>■ <b>No:</b> 1 record is created in the <b>History</b> table (amHistory) for the modifications brought about by importing the data, when these modifications concern a field or a link for which history is kept.</li> </ul>	Yes
Wizards	BASIC functions	This option stores Basic functions that can be called in <b>Wizard</b> type actions.	
Tips of the day	Text of tips	List of tips of the day displayed when AssetCenter is launched.	TIP2, "Shift+F9 starts the wizard debugger."

Section	Name of the option	Description	Example of values
LDAP authentication	Use an LDAP server for authentication	<ul style="list-style-type: none"> <li>■ <b>Yes:</b> The password entered when connecting to the database is looked up in an external LDAP directory.</li> <li>■ <b>No:</b> The password entered when connecting to the database is looked up in the <b>Password</b> field (LoginPassword) of the <b>Employees</b> table (amEmplDept).</li> </ul>	Yes
LDAP authentication	Authorize anonymous LDAP connections (empty login without password)	<ul style="list-style-type: none"> <li>■ <b>Yes:</b> Anonymous LDAP connections are authorized.</li> <li>■ <b>No:</b> Anonymous LDAP connection is disabled and the LDAP server checks login and password validity.</li> </ul> <p>This option is only taken into account if the <b>Use an LDAP server for authentication</b> option is set to <b>Yes</b>.</p>	Empty
LDAP authentication	Use a secure connection (SSL) to the LDAP server	<ul style="list-style-type: none"> <li>■ <b>Yes:</b> Authentication is performed via the LDAP certificate directory (<b>LDAP-CertDir</b> or <b>LDAPCertificate</b>).</li> <li>■ <b>No:</b> The password entered when connecting to the database is looked up in an external LDAP directory.</li> </ul> <p>This option is only taken into account if the <b>Use an LDAP server for authentication</b> option is set to <b>Yes</b>.</p>	Yes
LDAP authentication	LDAP server name	<p>Name of the server exposing the directory in which the AssetCenter user logins and passwords are stored.</p> <p>This option is only taken into account if the <b>Use an LDAP server for authentication</b> option is set to <b>Yes</b>.</p>	MyLDAPServer
LDAP authentication	LDAP server port	<p>LDAP server port used to listen for requests concerning the directory.</p> <p>This option is only taken into account if the <b>Use an LDAP server for authentication</b> option is set to <b>Yes</b>.</p>	389

Section	Name of the option	Description	Example of values
LDAP authentication	SQL name of the 'amEmplDept' table in which the LDAP directory login is stored.	<p>By default, this option supposes that you are using the same login in the AssetCenter database and in the LDAP directory: This login is then stored in the <b>Login</b> field (UserLogin) of the <b>Employees</b> table (amEmplDept).</p> <p>You may choose to differentiate between the LDAP directory login and that of AssetCenter. You must select a field to store the LDAP directory login and indicate its SQL name using this option.</p> <p>This option is only taken into account if the <b>Use an LDAP server for authentication</b> option is set to <b>Yes</b>.</p>	Field1
Authorization	License file	<p>License file that you inserted in the AssetCenter database (<b>Tools/ Edit the license file</b> menu).</p> <p>This option cannot be modified in this window.</p>	
Authorization	AssetCenter Server signature	<p>Used by AssetCenter to verify that AssetCenter Server regularly connects to the database.</p> <p>This option cannot be modified in this window.</p>	Updated by AssetCenter Server.
Authorization	Desktop Administration server signature	<p>Used by the Server Administration Console of Desktop Administration to verify whether Desktop Administration Server is connecting regularly to the database.</p> <p>This option cannot be modified in this window.</p>	Updated by Desktop Administration Server.
Authorization	Remote Control server signature	<p>Used by the Server Administration Console of Remote Control to verify whether Remote Control Server is connecting regularly to the database.</p> <p>This option cannot be modified in this window.</p>	Updated by Remote Control Server.

Section	Name of the option	Description	Example of values
Features	Disable recalculation of consolidated features	<ul style="list-style-type: none"> <li>■ <b>Yes:</b> The values of consolidated features are not be recalculated when the values of the features making them up change.</li> <li>■ <b>Yes:</b> The values of consolidated features are recalculated when the value of at least one of the features making them up changes.</li> </ul>	Yes
Controlling access to the Asset-Center database	Connection slot update delay (minutes)	Interval at which the client updates the login slot.	10
Access control	Enable automatic disconnection	Concerns the automatic disconnection (timeout) of a client from the database if inactive.	Yes
Access control	Slot timeout (seconds)	Timeout period for the database connection.  This option is only taken into account if the <b>Enable automatic disconnection</b> option is set to <b>Yes</b> .	1800
Documents	Maximum size of documents that can be inserted into the database (bytes)	This is maximum size of files that can be inserted in the <b>Documents</b> table (am-Document).	5 242 880
System data	Build	Build number of the the <b>System data</b> imported into the database.  The <b>system data</b> is automatically imported into the database when it is created.  This option cannot be modified in this window.	
Time zone	Data: Time difference in relation to the reference GMT time zone (hours)	Time zone of data in the database. The expected value is a numeric value specifying the difference with reference to GMT.  An empty value means that no time zone is used.  Zero (0) defines the time zone as GMT.	-2 (time zone GMT-2)

Section	Name of the option	Description	Example of values
Time zone	Server: Time difference in relation to the reference GMT time zone (hours)	Time zone of the server. The expected value is a numeric value specifying the difference with reference to GMT.  An empty value means that no time zone is used.  Zero (0) defines the time zone as GMT.	4 (time zone GMT+4)
Event management	Expiration time for output events (hours)	Output events are queues that can receive data and export it to external applications. The expiration time sets the value that determines when to purge these events.	336
Event management	Expiration time for input events (hours)	Input events are queues that can receive data and import it from external applications. The expiration time sets the value that determines when to purge these events.	336
Leasing	Authorize automatic asset LRF calculation	<ul style="list-style-type: none"> <li>■ <b>Yes:</b> The LRF field (pLRF) of assets is automatically calculated by AssetCenter depending on the values available.</li> <li>■ <b>No:</b> The LRF field (pLRF) of assets is not recalculated automatically. Its value is used as a constant to calculate the other values that depend on it.</li> </ul>	Yes
Password	Length of validity of the password in days		Yes
Password	Use password historization to stop passwords from being recycled	User passwords are archived in order to stop the same password from being used several times in a row by the same user.	Yes
Password	Number of days before expiration of the password	Delay in days before a system alert is raised.	Yes
Password	Name of library that validates the password formats.	Library defining the formats of user passwords.	Yes

Section	Name of the option	Description	Example of values
Full-text search	Enable full-text search	<p>By default, simple filters enable you to use the <b>LIKE</b> operator.</p> <p>This operator searches a string within the values of a field.</p> <p>This search is performed directly by AssetCenter.</p> <p>Certain DBMSs offer equivalent functionality with greater efficiency.</p> <p>When this is the case, this function must be specifically enabled at the DBMS level in order for it to be available.</p> <p>This search function then uses the <b>CONTAINS</b> operator instead of the <b>LIKE</b> operator.</p> <p>The <b>Enable full-text search</b> option enables the <b>CONTAINS</b> operator in the simple filter editor.</p>	Yes

---





# Indexes

& (special character), 41  
< (special character), 41

## A

Access control  
    Options, 101  
Access restrictions  
    Exclusions, 39  
Action buttons, 49  
Actions  
    Action buttons - creation, 49  
Activate modules (menu), 16  
Add a detail (menu), 48  
Add a field (menu), 45  
Add a page (menu), 50  
Add a parent link (menu), 47  
Add a table (menu), 45  
Add features (option), 45  
Add linked record (menu), 46  
Administration rights (option), 95  
Apply (button), 20  
AssetCenter Database Administrator  
    (See Also Customizing databases)  
AssetCenter Server signature (option), 100  
Authorization - options, 100

Authorize automatic asset LRF calculation  
(option), 102

## B

B (HTML tag), 41  
BASIC functions (option), 98  
Basic functions - options, 98  
Basic - options, 98  
bin (files), 66  
Build (option), 101  
Buttons, 42  
    Customization, 21

## C

Calculated fields  
    Default values, 39  
cfg (files), 66  
Configure object (menu), 42  
Configure tabs (menu), 19  
Connection - slots - options, 101  
Connection slot update delay (option), 101  
Consolidated features - options, 101 , 101  
Consumables - options, 98  
Counter names, 39  
Counters

- Default values, 39
- Names - special characters, 39
- Create expense lines for consumables as soon as received (option), 98
- Create view from current window (menu), 27
- Customization, 15-21
  - General options, 19
  - Modules, 15
  - Toolbar, 16
- Customize toolbar (menu), 16, 16
- Customizing databases, 33-63
  - Action buttons, 49
  - Buttons, 42
  - Details, 48, 42
  - Fields, 45
  - Fields, links and indexes, 36
  - Important note, 63
  - Index, 45
  - Links, 45
  - Modifications
    - Backing up, 62
    - Control, 63
    - Languages, 62
  - Objects, 44, 33
  - Pages, 61, 50
  - Tables, 45, 35

## D

- Data: Time difference in relation to the reference GMT time zone (option), 101
- Database objects, 60
- Database options (See Databases)
- Database options (menu), 95
- Databases
  - (See Also Customizing databases)
  - Automatic disconnection -options, 101
  - Description, ?-41
  - Options, 95-103
    - Configuration, 95
    - Configuration - window, 96
    - Restrictions, 95
- Default (parameter), 38
- Default values
  - Access restrictions, 39

- Calculated fields, 39
- Counters, 39
- Description - errors, 40
- Links to the Comments table, 39
- Desktop Administration server signature (option), 100
- Details
  - Customization, 42
  - Pages - adding, 61
- Detail screens (See Details)
- Detail windows (See Details)
- Disable history during import (option), 98
- Disable recalculation of consolidated features (option), 101
- Display languages
  - Adding objects to a database, 62
- Documents - options, 101
- Drop current view (menu), 28
- dsd (files), 66

## E

- Enable automatic disconnection (option), 101
- Enable full-text search (option), 103
- Execution error - Type error (error), 40
- Expense lines - options, 98
- Expiration time for input events (option), 102
- Expiration time for output events (option), 102

## F

- Field 'XXX' unknown in 'XXX' (error), 40
- Fields
  - (See Also Help on fields and links)
  - Customization, 36
  - Default value, 39
  - Insertion, 45
- Filtre (parameter), 37
- FONT COLOR (HTML tag), 41
- FONT FACE (HTML tag), 41
- FONT SIZE=+n (HTML tag), 41
- FONT SIZE=-n (HTML tag), 41
- Formatting (parameter), 37
- Full-text search - options, 103

**G**

General options, 19

**H**

Help on fields and links

Reserved characters, 41

Help on fields and links - recognized HTML tags

Editing, 40

Hierarchic tables, 47

Historized (parameter), 37

History - options, 98

HR (HTML tag), 41

HSplitControl (object separator), 60

**I**

I (HTML tag), 41

Icons (See Toolbar)

Import - options, 98

Index

Customization, 36

Insertion, 45

Input events - options, 102

Irrelevance (parameter), 37

**L**

LDAP - options, 99

LDAP server name (option), 99

LDAP server port (option), 99

Leasing - LRF - options, 102

Leasing - options, 102

Let AssetCenter Server create the items received in the portfolio (option), 97

LI (HTML tag), 41

License

File stored in the database, 100

License file (option), 100

License - options, 100

Links

(See Also Help on fields and links)

Customization, 36

Default value, 39

Insertion, 45

**M**

Mandatory (parameter), 38

Maximum size of documents that can be inserted into the database (option), 101

Modify current view (menu), 28

Modules, 15

**N**

Non breaking space (special character), 41

**O**

Output events - options, 102

**P**

Pages

Adding to a detail, 61

Creation, 50

Passwords - options, 99

Procurement - options, 97

**R**

Read only (parameter), 37

Receiving slips - options, 97

Reinitialize screens (menu), 15

Remote Control server signature (option), 100

Reset (button), 20

**S**

Save database configuration (menu), 42

Server: Time difference in relation to the reference GMT time zone (option), 102

Size (parameter), 36

Slot timeout (option), 101

Special characters, 39

SQL name of the 'amEmplDept' table in which the LDAP directory login is stored (option), 100

SQL names

Database objects - conflicts, 63

str (files), 66

stt (files), 66

System data

Options, 101

**T**

- Tables
  - Creation, 45
  - Customization, 35
- Tabs, 18
- Test databases, 63-67
  - Customization, 63
- Tests (See Test databases)
- Text of tips (option), 98
- Time zones, 80
- Time zones - options, 101
- Tips of the day - options, 98
- Toolbar
  - Buttons
    - Adding, 16
    - Deletion, 17
    - Moving, 17
  - Customization, 16
  - Default toolbar, 18
  - Position, 18
  - Separators
    - Adding, 17
    - Deletion, 17
- Tools (See Toolbar)

**U**

- Use an LDAP server for authentication (option), 99
- Use time zones (option), 80
- usr (files), 66

**V**

- Validity (parameter), 35
- Views, 25-29
  - Buttons - associating, 29
  - Creating, 27
  - Definition, 26
  - Deletion, 28
  - Displaying, 27
  - Editing, 29
  - List, 27
  - Modifying, 28
  - Query wizard, 27
- VSplitControl (object separator), 60

**W**

- wiz (files), 66
- Wizards - options, 98

**X**

- xml (files), 66



