Asset Manager

Software Version: 9.61

Windows® and Linux® operating systems

Tailoring

Document Release Date: September 2016

Software Release Date: September 2016

Hewlett Packard Enterprise

Legal Notices

Warranty

The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© 1994 - 2016 Hewlett Packard Enterprise Development LP

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

This product includes an interface of the 'zlib' general purpose compression library, which is Copyright © 1995-2002 Jean-loup Gailly and Mark Adler.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: https://softwaresupport.hpe.com/.

This site requires that you register for an HPE Passport and to sign in. To register for an HPE Passport ID, click **Register** on the HPE Software Support site or click **Create an Account** on the HPE Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HPE sales representative for details.

Support

Visit the HPE Software Support site at: https://softwaresupport.hpe.com.

This website provides contact information and details about the products, services, and support that HPE Software offers.

HPE Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support website to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HPE support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HPE Passport user and to sign in. Many also require a support contract. To register for an HPE Passport ID, click **Register** on the HPE Support site or click **Create an Account** on the HPE Passport login page.

To find more information about access levels, go to: https://softwaresupport.hpe.com/web/softwaresupport/access-levels.

HPE Software Solutions Now accesses the HPE Software Solution and Integration Portal website. This site enables you to explore HPE Product Solutions to meet your business needs, includes a full list of Integrations between HPE Products, as well as a listing of ITIL Processes. The URL for this website is http://h20230.www2.hp.com/sc/solutions/index.jsp.

Contents

Customizing Databases 8
Customizing the database 8
Development best practices 9
Use a test functional domain to customize database objects 9
Tag the Web services 10
Checking the functional domains associated with Web service objects 15
Customizing existing objects 16
Precautions 18
Customizing tables 18
Customizing objects 19
Customizing a field, a link or an index 20
Customizing a screen 26
Customize a page: Insert HTML content 28
Creating new objects 34
Methods of creation 35
Creating a table 36
Creating a field, link or index 37
Tracking user changes to records 40
Creating a screen 41
Creating action buttons 44
Creating a page 45
Adding a page to a screen 55
Propagating customizations to other supported languages 55
Saving modifications 56
Verifying your modifications 56
Important note 57
Screen sets 57
Overview 58
Practical case 1: Using screen sets 61
Customization prepared in a test database 68

Copying your working database 69
Propagating customizations from a test database to a production database 71
Exporting the database structure 72
Importing the new database structure 73
Migrate the data using HPE Connect-It 74
Look out for software counters 74
Customization made by several operators at the same time 74
Explanation of the problem 75
Possible types of customizations 75
Recommended methodology 77
Global process 77
Exporting data and structure of the production database 78
Importing the data and structure of development databases 83
Customizing the development databases 88
Exporting the data and structure of development databases 89
Importing data and structure into the production database 90
References: Customizable tables 91
Database options 99
To configure the database options 100
Database option configuration window 101
Customizing Screens 111
Customizing screens 111
Definition of a screen 111
Differences between screens and views 112
To create a screen from scratch in Asset Manager Application Designer 113
To create a screen from a view in the Windows client 113
Managing filters in screens 114
To display a screen 115
To modify a screen 116
To delete a screen 116
Customizing Windows Clients 117
Customizing a Windows client 117
Activating the modules 118

Customizing the toolbar 118
Customizing tabs 120
General options 121
User defined customizations 123
User defined colors in scripted queries for charted statistics 124
Using Views 125
Definition of a view 126
Creating a view 128
Displaying a view (Windows client) 129
Deleting a view 129
Modifying a view 130
Editing a view 130
Associating a view with a button in a screen 130
Customizing Web Clients 132
Modifying the Web client's default behavior 133
Disabling authentication when accessing the WSDL 134
Changing the text and the image of the banner 136
Reducing the time it takes to display screens for the first time 137
Defining the number of lines to export to Excel 139
Defining the number of lines to display in the lists 139
Modifying text read by screen reader software 140
Defining the number of records to display in the drop-down lists 141
Choosing the location of the workspace toolbar 141
Modifying how often the cache managing the workflow schemes and instances is refreshed 142
Refreshing a detail screen when the cursor leaves a field's or link's edit zone 143
Modifying the delay before a link's list of values is refreshed 144
Using other parameters to generate the Asset Manager Web Tier log file 144
Using another screen from the Business home pages table to display business home pages 145
Modifying the style sheet to use to display and print HTML pages from Asset Manager Web Tier 146
Modifying how often the currency cache is refreshed 146

Modifying how often the itemized lists cache is refreshed 147
Enabling the confirmation page before a script type action is executed . 148
Modifying the maximum number of segments to display in report or statistics graphs 148
Modifying the maximum number of reports to display in dashboards 149
Modifying the maximum number of segments to display in graphs for business home pages 149
Modifying the maximum number of statistics to display in business home pages 150
Modifying the root used for URLs in Asset Manager Web Service 150
Modifying the root used for URLs in Asset Manager Web Service 151
Modifying the root used for URLs in Asset Manager Web Tier 152
Modifying the name of Asset Manager Web Service that is displayed in administration screens of your application server 152
Modifying the name of Asset Manager Web Tier that is displayed in administration screens of your application server 153
Creating a logon disclaimer for the Web client 154
User defined customizations 155
Preferences pane 155
Customizing Chart Colors 158
Configuring list columns 158
Editing the package.properties file 159
Homepage of the Web client 163
How to create a homepage widget 164
Creating an AQL widget 164
Adding an AQL widget 164
Configuring the AQL widget 165
Generating the AQL widget 172
Unit 172
Creating a script widget 172
Adding a script widget 172
Configuring the script 173
Data source of the script widget 174
Parameters of the script widget 175
Drill-down in the script widget 176

Writing a script to generate the JSON object 177
Personalizing the homepage 184
User profiles 184
Homepage widgets management 184
Homepage templates management 185
How to create a homepage template 186
Importing/exporting homepage widgets and templates 186
Importing/exporting homepage widgets 186
Importing/exporting homepage templates 188
Business home pages 189
What is a business home page? 190
How business home pages work 190
How to create or modify a business home page 191
Recommendations when creating business home pages 191
HTML syntax for useful links 191
Unfiltered Asset Manager lists 192
Asset Manager lists filtered by query 192
Asset Manager lists filtered on an indexed field 192
Asset Manager Wizards 193
External Web pages 193
Specifying default home pages and home pages based on user roles 193
Deleting a business home page 195
Send documentation feedback 196

Customizing Databases

This chapter introduces how to customize Asset Manager databases.

Customizing the database 8
Development best practices 9
Customizing existing objects 16
Creating new objects 34
Screen sets 57
Customization prepared in a test database 68
Customization made by several operators at the same time 74
Database options 99
To configure the database options 100
Database option configuration window 101

Customizing the database

Asset Manager Application Designer 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 Asset Manager 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 Asset Manager, 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 Asset Manager database (tables, fields, links, indexes, screens, and so on):

- Existing objects.
- New objects.

This section includes:

Development best practices 9

Customizing existing objects 16
Creating new objects 34
Screen sets 57
Customization prepared in a test database 68
Customization made by several operators at the same time 74

Development best practices

This section includes:

Use a test functional domain to customize database objects 9
Tag the Web services 10
Checking the functional domains associated with Web service objects 15

Use a test functional domain to customize database objects

Recommendation

We recommend that you associate a new object or a new version of an object (a page or screen, for example) with a test functional domain before making it visible.

You can also mask the test functional domain to keep objects that are being developed hidden from users.

Justifications

As long as an object is incorrect, the functional domain to which it is linked is not generated as a Web service for Asset Manager Web.

Consequently, all objects (even valid ones) linked to the functional domain disappear from the Web services exposed by Asset Manager Web.

Implementation

To associate an object to a test functional domain:

1. Start the Windows or Web client.
2. Display the functional domains (**Administration/ Functional domains** link on the navigation bar).
3. Create the test functional domain by clicking New.
   To ensure that no Web service is generated for this functional domain, populate the **Web Service** (seWebService) field with the value **none**.
   Note the SQL name of the test functional domain.
4. Start Asset Manager Application Designer.
5. Select the new object or the object to modify.
6. Link this object to the test functional domain by entering the SQL name of the test functional domain in the object's **Domain** field.
7. Once the object has been tested and validated, link it to the appropriate exposed functional domain.

Tag the Web services

How applications other than the Windows client access the Asset Manager database

The Web services let applications other than the Asset Manager Windows client to query or modify the contents of the Asset Manager database.

These applications (Asset Manager Web, or tailored applications) interact with the database via functions provided by the Web services.

These functions are described in a WSDL document.

> **Note:** WSDL (Web Services Description Language) is a language that describes how to access an application's Web services.
>
> A WSDL document describes each of the available services in XML format: Operations that can be performed, messages used, data types used, and communication protocols used.

> The WSDL document is generated on the fly (it is not saved as a viewable file).
>
> An application can read the WSDL document to determine which functions are available on the server.
>
> This application can then use SOAP to call one of the functions listed in the WSDL document.

The WSDL document exposes the current database functions as well as the functions that were available at specific moments if the database was tagged at those specific moments.

What are the prerequisites for accessing the WSDL document?

- A valid HP AutoPass license for Asset Manager must be installed to be able access the Web services.
  See Asset Manager **Administration** guide, chapter **Installing License Keys**.
- Asset Manager Automated Process Manager must be started and the **Signal presence of database server** module (UpdateToken) must be enabled.

What objects are manipulated by the Web services?

The Web services functions can manipulate the following objects:

- Fields, links, indexes and screens of the database's tables
- Script and wizard type actions
- Filter queries for database table records

What tag identifier is used for the current version of the Web services?

The current version is identified as **HEAD**.

Why tag the Web services?

Tagging is done in order to track the state of the objects manipulated by the Web services at a given moment in time.

Using tagging, you can force applications that access the database, but do not do so via the Windows client, to use objects that correspond to the state of the database at a given moment in time.

This enables you to avoid any problems such as the following:

You are creating new query filters

You create the queries in the current database.

These queries are transmitted to the Web services (WSDL document tagged **HEAD**).

If you configured Asset Manager Web to access the database via the Web services tagged as **HEAD**, users will have immediate access to the query that you have just created.

If execution time for this query is very long, you may hinder the performance of the Web client.

If you configured Asset Manager Web to access the database via the Web services that were tagged when the database was stable, users will be able to access the database in a more stable state.

How do I tag the Web services at a given moment in time?

Once the structure of the database is in a state that you want to save:

1. Start Asset Manager Application Designer.
2. Connect to the database (**File/ Open.../ Open existing database** menu).
3. Tag the Web services (**Action/ Tag web service...** menu).

   > **Note:** If you re-tag the Web services with an identifier that has already been used, the definition of the stored Web services is updated. If you recompile the code that uses the Web services in your development environment, the errors that are displayed will underscore the incompatibilities with the previous tagged version.

4. If any error messages indicate that inconsistencies exist between the objects manipulated by the Web services and the database structure, correct these inconsistencies and then try to tag the version again.
   Correcting inconsistencies requires you to perform one of the following operations:
   - Modify objects that are not consistent with the database structure
   - Modify the structure of the database that is not consistent with the objects
5. Perform the operations described in section How to ensure that the tagged Web services are not corrupt in relation to the database structure.
6. Perform the operations described in section How can make an application use another tagged version of the Web services?.

Tagging lets you track the state of the database in the WSDL document and assign an identifier to this state.

The Asset Manager Web Service instances generate a WSDL document automatically when they access the Asset Manager database.

How to ensure that the tagged Web services are not corrupt in relation to the database structure

Asset Manager provides an administrative tool for validating or debugging tagged Web services. This wizard checks customized features (such as pages, screens or wizards) for selected functional domains. As such, it is a recommended Best Practice to use it on a regular basis in conjunction with development of features for the Web client.

1. Start Asset Manager.
2. Connect to the database.
3. Start the **Check definitions of the Web services...** (sysCoreWebServiceCheck) wizard (**Administration/ Check definitions of the Web services...** link on the navigation bar).
4. Select one or more functional domains (from the list of stand-alone Web services).
5. Click **Finish**.
6. The wizard generates the Web service, analyzing all related elements and properties. If the list of messages contains any errors (preceded by the ⛔ icon), the corresponding object needs to be edited to correct the error.
7. Once all functional domains have been tested without producing errors, tag the Web services again. (See How do I tag the Web services at a given moment in time?).

Examples of errors that may be detected by the wizard:

- A button that is part of a screen referencing a non-existent action: Add the action or modify the button
- A page that contains a field or link that no longer exists: Add the field or link or modify the page
- For the same page, there are several instances of the `LinkEdit` control, and the same link is used by more than one of these instances: You need to distinguish them using the `ApiName` parameter
- An **AQL** type calculated field (supported by the Windows client but not the Web client)
- A page that is visible associated with a functional domain that is not exposed: In the detail of the functional domain or of its parent functional domains, the value of the **WEB service** (seWebService) fields prevents the functional domain from being exposed in the Web services.

How to ensure that the tagged Web services will operate properly

1. The operations in the following sections must have been performed without any errors, immediately one after the other (in the order that you choose):
   - How to ensure that the tagged Web services are not corrupt in relation to the database structure
   - How do I tag the Web services at a given moment in time?
2. Start an instance of the Web client and check to see if you can connect with a browser (navigator).

Where is tagged version saved?

All tagging information is stored in the **Configurations and preferences** (amSysConfig) table of the Asset Manager database.

The WSDL document can be automatically generated from this information when Asset Manager Web Service accesses the database.

How can make an application use another tagged version of the Web services?

Once you have tagged the Web services with Asset Manager Application Designer, if you want Asset Manager Application Designer to use this tagged version to access the Asset Manager database:

If you are using Asset Manager Web

Configure the Asset Manager Web Tier instances to communicate with the Asset Manager Web Service using the tagged version that you have chosen.

Modify the **WebService.Version** parameter in the **package.properties** by following the instructions in the **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

If you are using an application that you have developed

Generate new stubs from the given tagged version's WSDL document and modify your application as needed if it uses objects manipulated by the Web services that were modified.

In what instances can using a tagged version of the Web services be problematic?

If the changes to the database since the last tagging operation involve:

- The modification of a field type
- The deletion of a table, field, link, index, itemized list value, page, screen or query

And the Web service tries to access one of these objects, an error will occur and the user will be notified.

In what instances should the version be tagged as **HEAD**?

When you develop query filters or actions, for example, it is useful to use the **HEAD** tag in order for these queries or actions to be automatically available via the Web client for you to test them.

Displaying the WSDL document

1. Start Internet Explorer.
2. Go to the following URL:

   ```
   http://<Name or IP address of the Asset Manager Web Service
   server>:<Asset Manager Web Service port>/AssetManagerWebService
   ```

   For example: **http://MyServer:8080/AssetManagerWebService**
3. Internet Explorer displays available tagged versions, available functional domains for each version and operations that can be performed on the database for each functional domain.

Here is another, however less legible, way to view the tags:

1. Start the Asset Manager Windows client.
2. Connect to the database.
3. Display the **Configurations and preferences** (amSysConfig) table (**Administration/ List of screens...** menu).
4. Records whose **Section** (CfgSection) field is equal to **DocumentDesc** describe the functions of the Web services for each "functional domain/tag/language" group.

Checking the functional domains associated with Web service objects

Web service objects are associated with a functional domain.

Some of the objects are called when the database connection is established or when the associated screen is displayed. These objects are linked to a functional domain.

These objects call a screen or a wizard in the database and expect this screen or wizard to be from the same functional domain as the domain that is linked to the object.

If the screen or wizard is not found in the expected functional domain, an error occurs.

This happens if a screen or wizard is associated with another functional domain as the result of one of the following changes:

- In the detail of a functional domain, you modify the **WEB Service** (seWebService) or **SQL name** (SQLName) field.
- You modify a screen's **Domain** field using Asset Manager Application Designer.

You must also make the changes in Asset Manager Web Tier's **web.xml** file and redeploy Asset Manager Web Tier:

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Search for the SQL name of the screen that changed functional domains.

   > **Tip:** This SQL name is found right after the **AssetManager.WS.** string.

3. Modify the parameter that is found by replacing the previous SQL name used for the functional domain with the new SQL name.
   For example, replace:

   ```
   WebService.Service.Currency=Currency
   ```
   With:

   ```
   WebService.Service.Currency=Finance
   ```
4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Customizing existing objects

Asset Manager Application Designer authorizes only limited customizations of the existing database objects. In order to avoid any problems while using Asset Manager, the permissions on certain object values and objects are set to read-only.

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

Two distinct forms of database customization are handled by Asset Manager Application Designer:

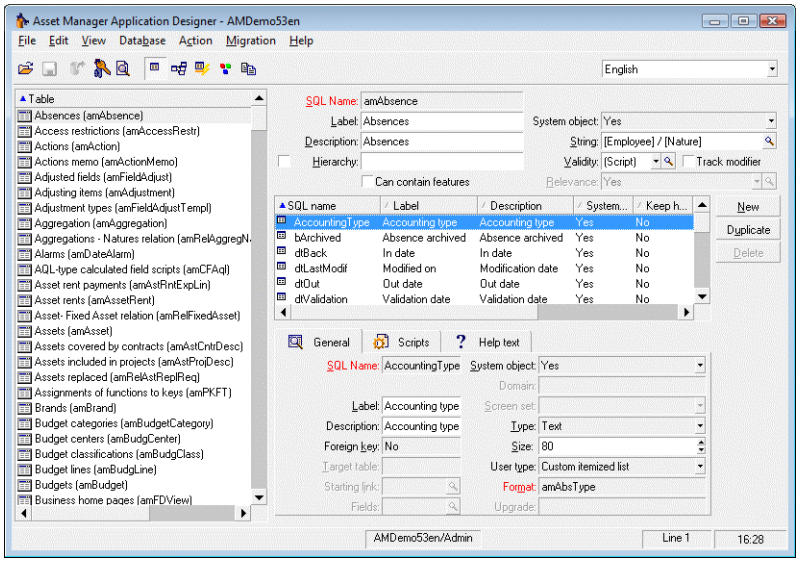
- Customization before creating the database.
- Customization after creating the database.

> **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.

Asset Manager Application Designer - main screen



This section includes:

Precautions 18
Customizing tables 18
Customizing objects 19

Precautions

If you modify the database structure, make sure you re-tag the Web services if you have deployed Asset Manager Web.

See Tag the Web services.

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 Asset Manager.
- The **String** field, which enables you to build the string which represents a record from this table in Asset Manager.
- 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, if 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 the 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.
- The **Track modifier** option enables you track who has modified a record in a particular table at what time and keep history of the information.
  See Tracking changes made on a record.

> **Note:** You cannot add custom fields or links to one of the following Asset Manager tables:
>
> - **amActionMemo**
> - **amCFAql**
> - **amComment**
> - **amFeatMemo**
> - **amFeatScript**
> - **amCatScriptData**
> - **amHelpMemo**
> - **amCFScript**
> - **amStatMemo**
> - **amWkEvtScript**
> - All Asset Manager relation tables (table names starting with "amRel"), for example, **amRelRestrProfile**, **amRelEscSchSev**.
>
> If you have added custom fields or links to these tables, some unexpected error may occur. It is recommended that you remove these custom fields or links.

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, Asset Manager Application Designer 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 that once in the table.
>
> The icon means that the indexed list of fields must not occur more that once in the table, except for the "NULL" value, which may be repeated any number of times.
>
> The represents a 1 link.
>
> The icon represents an n link.

Customizing a field, a link or an index

Customization with Asset Manager Application Designer

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 be accessed when customizing a database before creating it or when modifying a **gbbase*.*** 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 Asset Manager.
- 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:** Asset Manager 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 Asset Manager 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.

To be able to define an alarm for a date field:

1. Start Asset Manager Application Designer.
2. Connect to the database.
3. Select the table.
4. Select the page.
5. If you want to modify a system page, you must duplicate it first (system pages cannot be modified).
6. Click the **Contents** tab of the page's detail.
7. In the **List of fields** list, append the string **ComboAlarm-** to the beginning of the SQL name of the field.
   Example, if the SQL name of the field was **dStart**:

Change it to:

```
ComboAlarm-dStart
```

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, Asset Manager automatically inserts the number, incremented by one unit, for each new record.

  Counters are managed by the administrator using the **Administration/ System/ Counters** navigation 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.

  Asset Manager 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 Asset Manager 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 Asset Manager 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 Asset Manager.

> **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.

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.

HTML tags recognized in help on fields and links, continued

HTML Tag	Description
<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.

Reserved characters

Reserved character	Use the following tag to display this character
"<"	"<"
"&"	"&"
Non breaking space	" "

Customization with Asset Manager

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, and so on.

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

Customizing a screen

To customize a screen:

1. Select the table that the screen is part of.
2. Display the screens of the table (**View/ Screens** menu).
3. Select the screen to customize in the list.

You can modify:

In the **General** tab

- The **Name** and **Description** fields.

In the **List/Detail** tab

- Title of the screen
- Title of the list
- Columns in the list
- The other columns
- List/Detail proportions
- The QBE fields
- System filter
- Disable extended web filters

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
- 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 Asset Manager.
   - 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, and so on.>
     ```

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

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

     > **Note:** For **New**, **Duplicate** and **Delete** buttons, if they are made irrelevant, on the web client, they are invisible. For **Modify** and **Create** buttons, there are no corresponding buttons on the web client. If the **Modify** button is made irrelevant, on the web client, when you are modifying an existing record, the **Save** button is invisible. If the **Create** button is made irrelevant, on the web client, when you are creating a new record, the **Save** button is invisible.

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 **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.

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.

Customize a page: Insert HTML content

HTML content can be inserted into pages of the database tables.

This is made possible by **HtmlLabel** and **MultiLineHtmlLabel** class fields.

You can do this for both the Windows and Web clients.

How this works

The **HtmlLabel** and **MultiLineHtmlLabel** class fields are added to the components of one of the table's pages, and are not added to the table's list of fields and links.

In addition to where they are set in the page, the fields are defined by the following parameters:

Parameter	Parameter use
Class	• **HtmlLabel**: HTML object on a line • **MultiLineHtmlLabel**: HTML object on several lines
MainField	SQL name of the field or link that, when modified, causes the HTML object to be

Parameter	Parameter use
	refreshed
ExtField	SQL name of the calculated field that produces the HTML code of the object to display

Rendering of the HTML code depends on the client used to display it (Windows or Web).

- **Windows client**
  - The code is rendered by the Windows client
  - The list of recognized tags is limited
- **Web client**
  - The code is rendered by the Internet browser
  - The list of recognized tags corresponds to the HTML tags that are supported by the Internet browser and that have been validated for use with the HTML object.

Field example

Below is an example of an **HtmlLabel** class field:

Table	Computers (amComputer)
Page	Hardware (PgCpuGen)
Class	HtmlLabel
MainField	PhysicalAddress
ExtField	cf_sysCoreWebED

A **Hardware** (PgCpuGen) page exists in the **Computers** table.

This page contains an **HtmlLabel** class field whose value is calculated by the **Enterprise Discovery** (sysCoreWebED) calculated field. It is updated when the **Physical Address** (PhysicalAddress) field is modified.

This field is calculated by the following Basic scripts:

```
Dim strHtml as String
' strPhysAddr format is "AB-CD-EF-GH-IJ-KL"
Dim strPhysAddr as String
' strMAC format required by ED Web is "ABCDEFGHIJKL"
Dim strMAC as String
' Backend for ED stored in amOption (Section: 'Backends', Entry: 'ED')
```

```
' API amBackEndServerPath is dedicated to reading: amOption, 'Backends' section
Dim strServerED as String
strPhysAddr = [PhysicalAddress]
strServerED = amBackEndServerPath("ED")
if strServerED="" OR Len(strPhysAddr)<17 then
  strHtml = ""
else
  strMAC = Mid(strPhysAddr,1,2) + Mid(strPhysAddr,4,2) + Mid(strPhysAddr,7,2) + Mid
(strPhysAddr,10,2) + Mid(strPhysAddr,13,2) + Mid(strPhysAddr,16,2)
  strHtml = "<a href='"
  strHtml = strHtml + amBackEndServerPath("ED")
  strHtml = strHtml + "/nm/device/about.pcgi?device="
  strHtml = strHtml + strMAC
  strHtml = strHtml + ";device_type=MAC' target='_blank'>Display the detail in
Enterprise Discovery</a>"
end if
RetVal = strHtml
```

Example of HTML code produced by this script:

```
<a href='https://MyEnterpriseDiscoveryServerHostName/nm/
device/about.pcgi?device=000D6016CF6C;device_type=MAC'>Display the detail in
Enterprise Discovery</a>
```

To add an **HtmlLabel** and **MultiLineHtmlLabel** class field to a page

1. Start the Asset Manager Windows or Web client
2. Display the calculated fields (**Administration/ System/ Calculated fields** link on the navigation bar).
3. Create a new calculated field.
   Populate the fields and links in the following table:

Name of the field or link	SQL name of the field or link	Value
SQL name	SQLName	User defined SQL name **Note:** Remember this name. You will need it later.
Syntax tab		

Name of the field or link	SQL name of the field or link	Value
Table	TableName	Table to which the page containing the **HtmlLabel** or **MultiLineHtmlLabel** class field belongs.
Field type	seType	Basic script
Result type	seDataType	Text
Script of the calculated field (This label is not visible. The link is located in the **BASIC syntax** frame)	Script	Script that generates the HTML code to insert in the page

4. Start Asset Manager Application Designer.
5. Select the table that contains the page to modify.
6. Display the pages of the table (**View/ Pages** menu).
7. Select the page to modify.

   > **Note:** System pages cannot be modified (the value of the **System object** property is **Yes**).
   >
   > You can duplicate one if necessary.

8. Select the **Contents** tab.
9. In the edit zone at the top left of the **Contents** tab, enter the SQL name of the field that, when modified, triggers a refresh of the **HtmlLabel** or **MultiLineHtmlLabel** class field.
10. Place the cursor outside of the edit zone.
    A new object appears in the **List of fields** list.
11. Double-click the new object.
12. Populate the following properties:

Parameter	Parameter use
MainField	SQL name of the field or link that, when modified, causes the HTML object to be refreshed
Class	◦ **HtmlLabel**: HTML object on a line ◦ **MultiLineHtmlLabel**: HTML object on several lines
ExtField	SQL name of the calculated field that produces the HTML code of the object to

Parameter	Parameter use
	display. **Note:** You created the calculated field in a previous step. Warning: The SQL name of the field begins with **cf_**.

Recognized HTML tags

When you generate the HTML code using the calculated field, you will use HTML tags.

The following tags are recognized by both the Windows and Web clients:

Tag	Use	Attributes
<B> </B>	Displays text in bold	
<STRONG> </STRONG>	Displays text in bold	
<I> </I>	Displays text in italics	
<EM> </EM>	Displays text in italics	
<U> </U>	Underlines text	
 	Adds a line break	
<P> </P>	Adds a paragraph	
<HR>	Inserts a horizontal line	
<UL> </UL>	Used for an unordered list	
<LI> </LI>	Used for a list entry	
<CENTER> </CENTER>	Centers the text	

Tag	Use	Attributes	
<PRE>	Preformats the text		
<SMALL> </SMALL>	Decreases text size		
<FONT> </FONT>	Defines the font used to display the text	COLOR=	Text color
		SIZE=+	Increases text size
		SIZE=-	Decreases text size
		FACE=	List of comma separated fonts that the user agent should use by order of preference.
<ALIGN> </ALIGN>	Defines text alignment	CENTER	Centers the text
<A> </A>	Anchor	HREF=	Specifies the location of a Web resource. Defines a link between the current element (source anchor) and the destination anchor defined by the **HREF** attribute.

You can use other HTML tags and attributes if they are needed to appropriately use the HTML object.

The Internet browser will be able to render them correctly.

However, the Windows client will not.

Simplified syntax

If you want to store an **http**, **https** or **mailto** link type in a field and the link's text corresponds to the field label, you can use a simplified procedure:

Configure the database

1. Start Asset Manager Application Designer.
2. Select the table containing the page that will display the hypertext link.
3. Display the pages of the table (**View/ Pages** menu).
4. Select the page to modify.

   > **Note:** System pages cannot be modified (the value of the **System object** property is **Yes**).
   >
   > You can duplicate one if necessary.

5. Select the **Contents** tab.

6. In the edit zone at the top left of the **Contents** tab, enter the SQL name of the field that is used to store the URL address.
7. Place the cursor outside of the edit zone.
   A new object appears in the **List of fields** list.
8. Double-click the new object.
9. Populate the following properties:

Parameter	Parameter use
MainField	SQL name of the field used to store the URL address
Class	HtmlLabel

10. Save the changes to the database structure (**File/ Update database structure...** menu).

Populate the database

1. Start the Asset Manager Windows or Web client.
2. Display the record's detail.
3. Populate the reference field.
4. Save the information.

Use the URL address

1. Start the Asset Manager Windows or Web client.
2. Display the record's detail.
3. Display the tab containing the **HtmlLabel** class field.
   The **HtmlLabel** class field displays the label of the reference field as a hypertext link.
   When you click the link:
   - Windows client: The default Internet browser starts and displays the page corresponding to the URL address associated with the link.
   - Web client: The Web client displays the page corresponding to the URL address associated with the link.

Creating new objects

Asset Manager Application Designer enables you to create new objects for the database.

> **Caution:** 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.

This section includes:

Methods of creation 35
Creating a table 36
Creating a field, link or index 37
Tracking user changes to records 40
Creating a screen 41
Creating action buttons 44
Creating a page 45
Adding a page to a screen 55
Propagating customizations to other supported languages 55
Saving modifications 56
Verifying your modifications 56
Important note 57

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.
- Track user changes to records
- Create screens for your table.
- Create, if you want, action buttons present in the screen.
- Create the pages of your screens.
- Add the pages to your screens.
- Propagate customizations to other supported languages

- Save your modifications.
- Verify your modifications.

Creating a table

To create a new table:

1. Select the **Database/Add a table** menu.
2. Asset Manager Application Designer displays a creation window.
3. 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 Asset Manager.
   - 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. Asset Manager Application Designer will automatically create the additional tables necessary to deal with the features.
4. Click **Create**. Asset Manager 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.

> **Note:** For each table, you should create at least one screen with the exact same SQL name as the table (case sensitive).
>
> You can have multiple screens with the same SQL name if they are associated to different screen sets.
>
> The screens with the same SQL name as the table are used by Web services, because Web services do not work with tables but only with screens.
>
> Example: a wizard query searches the **amAsset** table; the Web service interprets this as a screen SQL name, not a table SQL name.

See **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. Asset Manager Application Designer 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.

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

To create a link:

1. Select the **Database/Add a link** menu.
2. Asset Manager Application Designer displays a creation window.
3. In this window, populate the following fields:\
   - **Link type**
   - For the source table:
     - **SQL name**
     - **Label**
     - **Description**
   - For the destination table:
     - **Table**: Indicate the destination table.
     - **SQL name**
     - **Label**
     - **Description**
4. Select the **Create an index for this link** option if you want to create an index.

See **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: RelNN, 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.

> **Note:** You should not add new fields or links to the following tables:
>
> - amActionMemo
> - amCFAql

- amComment
- amFeatMemo
- amFeatScript
- amCatScriptData
- amHelpMemo
- amCFScript
- amStatMemo
- amWkEvtScript

If you have added fields or links to these tables, you need to manually remove them to avoid unexpectedly errors.

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. Asset Manager Application Designer 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:

   ```
   lParentId,InternalRef
   ```

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

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

Tracking user changes to records

Adding the **LastModifier** link to tables

By default, the name of the user modifying a record is not stored in any of the tables.

To enable this option for a given table:

1. Start Asset Manager Application Designer.
2. Connect to the database.
3. Select the table.
4. Select the **Database/ Add a 'Last Modifier' link** menu item.
   This adds the **LastModifier** (LastModifier) link to the table.
5. Save the changes (**File/ Update database structure...** menu).

To enable this option for all tables:

1. Start Asset Manager Application Designer.
2. Connect to the database.
3. Select the **Database/ Add a 'Last Modifier' link to all tables** menu item.
   This adds the **LastModifier** (LastModifier) link to all tables.
4. Save the changes (**File/ Update database structure...** menu).

Resulting Consequences: When a user modifies a record, the **LastModifier** link is populated automatically.

If you want the **LastModifier** link to be visible in the detail of a table's records: See Creating a page.

Tracking changes made on a record

There are two ways by which you can track changes made to records of a table, depending on the level of granularity desired:

> **Note:** The change history can be checked from the **History** tab in the detail window of the record.

- On the individual field level

  For each field of a table, you can define whether to keep a history of modifications. The values for the field both before and after the modification are kept along with the modification time and the modifier information.
  Refer the **Advanced use** guide, chapter **Historization**.

- On the record level

  You can track modifications made on records of a table by selecting the **Track modifier** option for the table concerned using Asset Manager Application Designer. The option is located at the upper section of the main pane.
  This allows you to know who has made changes to a record at what time: each time you make any change to any record in the table, a line item is generated in the **History** tab of record's detail window. Information such as last modification time is recorded. The details of the changes are not recorded.

  > **Note:** By default, the **author** field is not added in the **History** tab. You need to add the field to the tab using the **Configure list...** contextual menu to view the last modifier information.

  > **Tip:** The line items for the record history can be distinguished from those for individual field history by the value of the **Field or link** column: they have a special value of **[Creation]** or **[Track modifier]** as opposed to specific field names without enclosing brackets.

Creating a screen

A screen is a graphical view of the information stored in the records of a table. Examples: screens are displayed in Asset Manager when you select a menu or click a toolbar icon. Screens are composed of one or more pages regrouped by tabs in the graphical interface of Asset Manager.

> **Note:** Asset Manager is provided with a set of standard screens that are used to support the functions of the application. Some screen parameters cannot be modified (example: The domain and the screen set associated with the screen).
>
> Generally, by default, each table is associated with a full and simple screen.

To create a screen:

1. Run Asset Manager Application Designer and open the database.
2. In the left-hand (**Table**) pane, select the table on which you want to create a new screen.

3. Select the **View/ Screens** menu or click the button on the toolbar.
4. Select the **Database/ Add a screen** menu, or click **New**.
5. Populate the following fields in the **General** tab:
   - **SQL name:** SQL name of the screen. This name enables you to:
     - Create a link between screens with the same nature that are associated with different screen sets.
     - Uniquely identify groups of screens and reference them in scripts or queries.

     The screen SQL Name must start with an alphabetical ASCII character (a-z and A-Z, no special characters are allowed).
     The _ character is allowed.
     For each table, you should create at least one screen with the exact same SQL name as the table (case sensitive).
     You can have multiple screens with the same SQL name if they are associated to different screen sets.
     The screens with the same SQL name as the table are used by Web services, because Web services do not work with tables but only with screens.
     Example: a wizard query searches the **amAsset** table; the Web service interprets this as a screen SQL name, not a table SQL name.
   - **Label:** Label of the screen.
     Do not use strings that can be confused with HTML tags (tags starting with **<**).
   - **Domain:** functional domain of this screen. 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 navigation bar under the corresponding functional domain.
   - **Screen set**: Screen set associated with the screen. The value of this field is used by Asset Manager to identify which screen to display depending on the screen set priority order associated with the Asset Manager user.

     > **Note:** For users with administrator rights, this is the order defined via the Windows client **Administration/ Database options** menu, **Screen sets** section, **Default order** option.
     >
     > For other users, this is the order in the user profile, **General** tab **Screen sets** field defined using the Windows or Web clients. If this field is not populated, Asset Manager uses the order defined via the **Administration/ Database options** menu, **Screen sets** section, **Default order** option in the Windows client.
6. Populate the following fields in the **List/Detail** tab:

- **Title of the screen:** Defines the title of the window displayed in Asset Manager. The **String** of the description is added to this title.
- **Title of the list:** defines the title of the page as it appears in the list of screens (**Administration/ List of screens**, in Asset Manager).
- **Columns in the list:** This field enables you to define the columns that will be displayed by default in the list window. The following syntax is used for this field:

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

  > **Note:** On the Windows client, the total width of the list dynamically changes with the size of the window. On the web client, the total width of the list is a fixed value (1,024 pixels).

- **Other columns:** This field lets you specify the list of columns that users can add to a list when they are associated with a user role whose **Authorize display of all fields and links in the lists** (bFullListCfg) check box is not selected.

  > **Note:** Users associated with a user role whose **Authorize display of all fields and links in the lists** (bFullListCfg) check box is selected are not limited in the selection of columns that they can add to a list.

  The following syntax is used:

  ```
  <SQL name of the field or link for column 1>,<SQL name of the field or link for column 2>,...
  ```

- **List/detail proportions:** ratio between the detail and the list.
- **QBE fields:** Populate this field to:
  - Windows client: Have a query wizard displayed before the screen is displayed (generated on the fly using information from the **QBE fields** fields).
  - Web client: To filter records from a list using the query wizard defined by Asset Manager Application Designer on the screen to which the list belongs, select the **QBE filter** from the drop-down list above the records list.

    Populate this field by providing the list of fields to display in the query wizard or filter. Separate fields with a comma.
    Fields will be displayed from bottom to top.
    Example for the **Work orders** table (amWorkOrder): **WONo,seStatus,Title,dtResolLimit**.

    > **Note:** In the Windows client, users select to use or not to use query wizards via the **Enable query wizards** option (**Edit/ Options/ Navigation** menu)

- **Query filter** and **System filter**: specify any default filters to apply to data displayed by the screen.
- Ignore user customizations: if this option is checked, the screen will behave as a **View**. Any changes that a user makes to the screen (column order, and so on), will be discarded when the screen is closed (Windows client only).

7. Click **Create**.

> **Note:** The screen will be visible under the selected domain in the Asset Manager Windows or Web client navigation bar when users next connect to the database.

You have created an empty shell for 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

Asset Manager Application Designer offers you to possibility of creating buttons in your screens. These buttons are used to trigger an action, display a screen, display a view (this option is not supported by the Web client) or print a report.

To create this button:

1. Select the screen to which you want to add a button.
2. Select the **Buttons** tab of the screen.
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 Asset Manager 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, and so on.>
  ```

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

  - **A** for an action.
  - **S** for a screen.
  - **V** for a view.

    > **Caution:** This option is not supported by the Web client.

  - **R** for a report.
- **Relevance**: enables you to specify the button's display parameters (yes, no or script).

  If you write script for this field, we recommend that you write an "IF" block in a single line. For example, use:

  ```
  RetVal = ("" = [Status] OR "construction" = [Status])
  ```

  Instead of :

  ```
  RetVal = FALSE
  If "" = [Status] OR "construction" = [Status] Then
    RetVal = TRUE
  End If
  ```

5. Click **Modify** to validate your changes.

Creating a page

In Asset Manager, 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 Asset Manager 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**).

     > **Note:** When adding a remote field (field/link from a linked table) to the page, a warning message **"As remote fields are not supported in screen pages, it is recommended to use a path to a remote page."** will be displayed. Ignore the warning by clicking **OK** and proceed.
     >
     > The remote field when displayed on the target page is not editable; modifications to the field values will not be saved.

     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 Asset Manager. 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 horizonal 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.

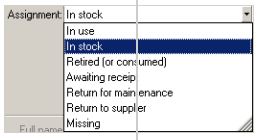
The expected syntax is the following:

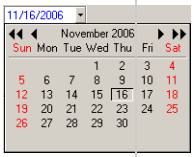
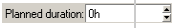
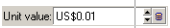
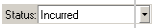
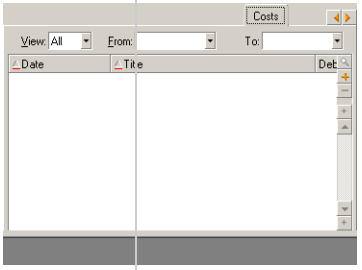
```
[<Control type>-]<SQL name of the object>[|<SQL name of a secondary object>]
[|<Property of the control>="<Value>"*]
```

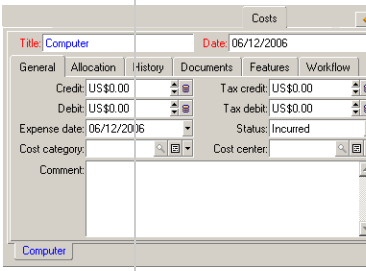
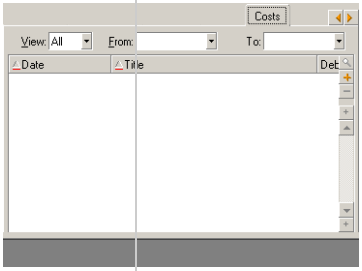
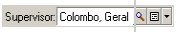
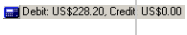
> **Note:** The SQL name of a secondary object is only used for certain properties.

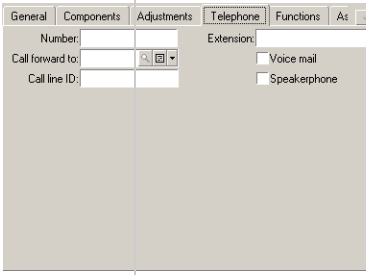
Types of graphic controls

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, multi-line. **Note:** By default, the label of the control is not displayed. To enable the label, use the **UseLabel** property and set its value to **1**. See Graphical control properties	
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.	

Control type	Description	Graphical example
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.	
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.	

Control type	Description	Graphical example
Notebook	Tabbed linked-records display control	
OneToManyLinkEdit	Linked records display control, enabling list view or tabbed view depending on user choice.	
LinkEdit	Selection control for a record in a list. **Caution:** If you insert several instances of the `LinkEdit` control in the same page, and the same link is used by more than one of these instances, then you must use the `ApiName` parameter to differentiate between them: See Graphical control properties	
ReversePasswordEdit	Password-entry control. The encryption uses a reversible method thus enabling Asset Manager to use it to authenticate itself to external systems.	
PasswordEdit	Password-entry control. This password is not reversible. Only Asset Manager 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.	

Control type	Description	Graphical example
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.	
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.	
DbPathCombo	Selection control for a field 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
Caption	The behavior of this property depends on the type of control: • For OneToManyLinkEdit, LinkEdit, LinkList controls: The property defines the title of the selection window of the linked record. • For Label, CaptionedLabel and MultilineLabel controls: The property displays a label whose value is that of a field in the table. • For the QueryCalc control: The property defines the label explaining the calculated value.	Label, CaptionedLabel,MultilineLabel, QueryCalc, LinkList, OneToManyLinkEdit, LinkEdit
Query	AQL query executed when the user clicks the icon.	QueryCalc
ListConfig	This property defines the columns in the list. The format used is the following: `ListConfig="<SQL name of the object>,<Size of the column>[,]*` Example:	LinkList, OneToManyLinkEdit

Property	Description	Control(s) using the property
	`ListConfig="Name,40,seType,30"`	
WhereCond	This property contains the Where clause of an AQL query. It enables you to filter the list of records in the drop-down list. For example, in the list of departments and employees, the **Supervisor** (Manager) link is associated with a control with the following WhereCond property: `WhereCond="bDepartment=0"` Thus the departments will not appear in the list and will not be available for selection for the value of the supervisor.	LinkList, LinkEdit
DistantDet	• SQL name of the detail displayed when viewing the detail of the linked record (icon) for the OneToManyLinkEdit and LinkEdit controls. • SQL name of the distant detail displayed for the DistantDet control. **Note:** The filter of this detail is applied as well as the WhereCond property.	OneToManyLinkEdit, DistantDet, LinkEdit
AddMode	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: • 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. • If the property is set to 0, the control is in selection mode only and the user must use the **New** button if they wish to create a record.	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. **Note:** If the action has a context table, the context table must be set to the source table of the link.	OneToManyLinkEdit
OnRemoveLink	This property contains the SQL name of an action to execute when the user clicks the button.	OneToManyLinkEdit

Property	Description	Control(s) using the property
	Note: If the action has a context table, the context table must be set to the target table of the link.	
Mode	This property defines the default display mode for this type of control: • If the property is set to 1, the control is displayed in list mode by default. • If the property is set to 2, the control is displayed in tab mode by default.	OneToManyLinkEdit
DescString	This property enables you force the description string of the window that presents the detail of the selected record (by clicking the button). The format used by this property is exactly the same as that of a description string. **Note:** For further information on description strings, refer to the section **Table description strings** in the **Administration** guide.	LinkEdit
ReadOnlyDetail	This property enables you to define the display mode of the detail window of the selected window: • If the property is set to 1, the detail is opened in read-only mode. • If the property is set to 0, the open detail may be modified by the user.	LinkEdit
AllowNone	This property enables you to authorize or deny selection of the **No table** value in the selection list. • If the property is set to 0, the value may not be selected. • If the property is set to 1, the value may be selected.	TableChoiceList
AllowNoMainField	This properyy enables you to display relationship (intermediary) tables (such as **amRelxxx**). • If the property is set to 0, the relationship tables are not displayed. • If the property is set to 1, the relationship tables are displayed.	TableChoiceList
OnlyFeature	This property enables you to display (or not display) the	TableChoiceList

Property	Description	Control(s) using the property
	tables that have a corresponding table of feature values (such as **amFVxxx**). • If the property is set to 0, all tables are displayed. • If the property is set to 1, only the tables with a corresponding table of feature values are displayed.	
NoFeature	This property enables you to display the feature value tables in the selection list. • If the property is set to 0, all tables are displayed. • If the property is set to 1, the feature value tables are not displayed.	TableChoiceList
HideSystem	This property enables you to restrict the list of table to those included in the user license. • If the property is set to 0, all tables are displayed. • If the property is set to 1, only the tables authorized by the license are displayed.	TableChoiceList
ApiName	This property is used when a page contains the same link more than once (generally, these links will have a different `WhereCond` property). The `ApiName` property is used by the Web services to distinguish between the different instances of this link.	LinkList

Caution: Some controls are not supported in the Web client. The list is given in the **Asset Manager Web Implementation** guide, **Differences between the Windows client and the Web client** chapter, **Differences between the clients/ Controls** section.

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 Asset Manager, 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.

Default IDs

The **HSplitControl** and **VSplitControl** separators, as well as the fields and links listed for a page, all have default IDs. Examples:

```
HSplitControl_0|Id="Group00"
```

```
Reason|Id="Field01"
```

Asset Manager uses these IDS for hidden processes. You cannot make any use of them.

Adding a page to a screen

To add pages to a screen:

1. Select the screen to which you want to add pages.
2. Select the **Pages** tab of the screen. You can now select the page(s) that you want to add to the screen. To do this:
   - Select in the list to the left (**Available pages**) the page(s) that you want to add to the screen, 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 screen, 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, and so on.

Propagating customizations to other supported languages

Objects added to the database are available in the language selected in Asset Manager Application Designer during the modification.

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

- Screens
- Pages

> **Tip:** You must therefore successively select each language to be supported (list on the right side of the toolbar in Asset Manager Application Designer) 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 Asset Manager Application Designer).
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/ Update database structure** or click .
2. Follow the instructions given by the wizard.

Verifying your modifications

To check your modifications, you need to launch Asset Manager 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 screen in the list that appears on screen and click **OK**. Your new screen 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 on the navigation bar of a Windows client.
5. Select a functional **Domain**. The **Name** of your view will appear under this functional domain on the navigation bar.
6. You can now directly access your new view or screen.

Important note

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

- View
- Screen

For example, if you create a new screen for the **Employees and departments** table (amEmplDept) and you associate a view whose SQL name is amEmplDept to this screen, your new screen, and not the amEmplDept table, will be displayed when you select the **Organization management/ Organization/ Employees and departments** link on the navigation bar.

Screen sets

This section includes:

Overview 58
Practical case 1: Using screen sets 61

Overview

Why are screen sets used?

Screen sets were introduced in Asset Manager to provide simplified screens tailored to each individual user as opposed to providing only more complex, full screens to all users.

Different versions of the same screen can be created using screen sets. For example: A full version and a simple version.

Users access the correct screen set through their user profile. For example: If simple screen sets are selected for a given user, Asset Manager will display a simple screen each time one is available for a given table.

How do screen sets operate?

Different versions of the same screen must use the same SQL name and be associated with a different screen set.

In the Asset Manager database, screen sets are given a default priority order (database options).

For each user profile, a priority order can be defined for screen sets that supersedes the database options.

The priority order for screen sets depends on user rights, functional rights and user profile access restrictions.

How do I implement screen sets?

The following table explains what you need to do to implement and set up screen sets:

Action to perform	Use
If necessary, the administrator creates any missing **functional domains** (via the Windows or Web client)	Functional domains are used to group screens in the Windows and Web client navigation bar. For example: If the **Screen** screen is associated with the **Domain** functional domain, the navigation bar displays the **Domain/ Screen** tree structure.
The administrator modifies	For each table in the database, the administrator can define

Action to perform	Use
existing **screens** or adds new ones (using Asset Manager Application Designer)	several different screens depending on the needs of each individual user. By default, most tables are defined with a full and simple screen. For example, if: • Two screens with the same SQL name **Screen** are associated with screen sets **Set 1** and **Set 2**. • The screen set **Set 1** is associated with **User 1** via the user profile. • The screen set **Set 2** is associated with **User 2** via the user profile. Doing the same using the Windows or Web clients **User 1** will display the screen associated with **Set 1** and **User 2** will display the screen associated with **Set 2**.
The Asset Manager database administrator sets the default **database options** parameters (using the Windows client).	The **Default order** option in the **Screen sets** section can be set in the database options. This option defines the list of screen sets to use starting with the highest priority first (example: **Full screens, Simple screens**). This option is used when the following user types request a table be displayed: • Users associated with a profile whose **Screen sets** (ScreenSets) field is empty • Users associated with a profile whose **Administration rights** (bAdminRight) option is selected When these users request a table be displayed, Asset Manager takes the identifiers of the screen sets in the order specified by the option and searches the table to see if it contains a screen associated with this screen set. When a screen is found, Asset Manager uses it to display the table. This default behavior can be superseded by the functional rights associated with the profile: Access to a given screen can be denied by the functional rights.
The Asset Manager database administrator defines **functional rights** (using the Windows or Web client)	Functional rights are used to authorize users access to screens and pages via the menu bar, the navigation bar and wizards. To determine which screens are displayed when a user connects to Asset Manager, Asset Manager checks the

Action to perform	Use
	following: 1. The user is described by a record in the **Employees and departments** (amEmplDept) table. 2. This record is linked to a user profile (**Profile** (Profile) link). 3. The profile is linked to functional rights (**Functional rights** link (FuncRights)). 4. The list of available function domains can be found in the detail of functional rights. Available screen sets can be found in each functional domain. Screens linked to the concerned functional domain and screen sets can are found within these screen sets. The detail of a functional right is used to grant access rights to various screens.
The Asset Manager database administrator defines **user profiles** (using the Windows or Web client)	User profiles are used to define user access rules for the Asset Manager database. Results for the screens depends on the following elements: • **General** tab, **Screen sets** field(ScreenSets): This field is used when a user associated with the profile requests a table be displayed. Asset Manager takes the identifiers of the screen sets in the order specified by the **Screen sets** field and searches the table to see if it contains a screen associated with this screen set. When a screen is found, Asset Manager uses it to display the table. This default behavior can be superseded by the functional rights associated with the profile: Access to a given screen can be denied by the functional rights. • **User rights** tab, **User rights** link (UserRights): Access rights to tables (create, delete, read, enter during creation, enter during update). • **User rights** tab, **User rights** link (UserRights): Access rights to screens and pages.

Operations that access screen sets

Screen sets are used by the following operations:

- Menus that display a table (example: **Portfolio/ Portfolio items...**)
- Windows and Web client navigation bar links that display a table (For example: **Asset lifecycle/ Supplier contract management/ Contracts**)
- Wizards that display screens via the **AmOpenScreen** and **AmOpenScreenEx** APIs.

Default configuration of the Asset Manager database

In the default structure of the Asset Manager database, most tables are defined with:

- A full screen associated with the **Full** screen set
- A simple screen associated with the **Simple** screen set

The default value of the **Screen sets/ Default order** database option is **Custom,MigratedSys,Full,Simple**.

You can also create **Custom** screen sets tailored to your needs.

Practical case 1: Using screen sets

Presentation

This practical case is intended to help you understand how to configure different screen sets for different users in Asset Manager.

Prerequisites

Before beginning this practical case, install the following applications and components:

- Microsoft SQL Server 2005
- Asset Manager
- Demonstration database
- Asset Manager Application Designer

Scenario

Let's look at the following scenario:

- Asset Manager database administrators require access to all fields and links for all tables when they display records.
- The Human Resources Advisor is responsible for updating the **Employees and departments (amEmplDept)** table. To do this, they require access to all fields and links for that table.
- The Human Resources Manager only wants to have access to an executive summary of the employee information.

Steps to follow

1. Create the functional domain **CP1 - Human Resources**

   The screens that you are going to create in the **Employees and departments** table will be linked to the newly created functional domain.

   a. Start the Asset Manager Windows client.

   b. Connect to the demonstration database selecting **Admin** as the user:

Field	Value
Login	Admin
Password	The password is empty

   c. Display the functional domains (**Administration/ Functional domains** menu).

   d. Create the following record using the fields and links as shown in the following table:

Field or link	Value
Name (Name)	CP1-Human Resources
SQL name (SQLName)	CP1_Human_Resources

   e. Exit Asset Manager.

2. Create the screens

   The screens will allow users to display a customized view of the **Employees and departments** table.

   a. Start Asset Manager Application Designer.

   b. Connect to the demonstration database selecting **Admin** as the user.

   c. Display the detail of the **Employees and departments** (amEmplDept) table.

d. Display the list of screens (**View/ Screens** menu).

e. Create the following screens by populating the fields and links as shown in the following tables (**New** button):

Field	Value
General tab	
SQL name	CP1_amEmplDept
Domain	CP1_Human_Resources
Screen set	CP1-Simplified
List/Detail tab	
Columns in the list	Name,20,FirstName,20,Title,30
Pages tab	
Pages in detail list	Add the page **sysamEmplDept**

Field	Value
General tab	
SQL name	CP1_amEmplDept
Domain	CP1_Human_Resources
Screen set	CP1-Complete
List/Detail tab	
Columns in the list	Name,20,FirstName,20,Title,30
Pages tab	
Pages in detail list	Add all pages of the **Available pages** list.

f. Start and enter the required information in the database structure update wizard (**File/ Update database structure** menu).

g. Exit Asset Manager Application Designer.

3. Enter the parameters for the database options

   The option that you are going to configure is used to define the order in which screen sets are made available to the administrator.

a. Start the Asset Manager Windows client.

b. Connect to the demonstration database selecting **Admin** as the user.

c. Display the database options via the **Administration/Database options** menu.

d. Populate the following option (**Current value** column):

Section	Option	Value
Screen sets	Default order	CP1-Complete,CP1-Simplified,Custom,MigratedSys,Full,Simple

e. Close all screens.

f. Reconnect to the database as **Admin** so that all database options are taken into account (**File/ Connect to database** menu).

4. Create the functional right **CP1-Human Resources**;

   The functional right that you are creating will be associated with the Human Resources Advisor and Human Resources Manager user profiles.

   a. Display the functional rights (**Administration/ Rights/ Functional rights** navigation menu).

   b. Create the following record using the fields and links as shown in the following table:

Field or link	Value
Description (Description)	CP1-Human Resources
SQL name (SQLName)	CP1_Human_Resources
Granted by default (bDefault)	Check this option
List of items	Make sure the **Value of parent** option is checked for all items.

   c. Close all screens.

5. Create the user right **CP1-Human Resources**

   The user right that you are creating will be associated with the Human Resources Advisor and Human Resources Manager user profiles.

   a. Display the user rights (**Administration/ Rights/ Users rights** menu).

   b. Create the following record using the fields and links as shown in the following table:

Field or link	Value
Description (Description)	CP1-Human Resources
Rights tab	
Object list	
Select all objects	
Read	Select this option
Enter during creation	Select this option
Enter during update	Select this option
Create	Select this option
Delete	Select this option

c. Close all screens.

6. Create the **CP1 - Human Resources Advisor** and **CP1 - Human Resources Manager** user profiles

   Access profiles will be assigned to the Human Resources Advisor and the Human Resources Manager.

   a. Display the user profiles (**Administration/ Rights/ User profiles** menu).

   b. Create the following records by populating the fields and links as shown in the following tables:

Field or link	Value
Description (Description)	CP1 - Human Resources Advisor
General tab	
Screen sets (ScreenSets)	CP1-Complete, CP1-Simplified
User rights tab	
User rights (UserRights)	Add the user right **CP1-Human Resources**
Functional Rights tab	
Functional rights (FuncRights)	Add the functional right **CP1-Human Resources**

Field or link	Value
Description (Description)	CP1 - Human Resources Manager
General tab	
Screen sets (ScreenSets)	CP1-Simplified, CP1-Complete
User rights tab	
User rights (UserRights)	Add the user right **CP1-Human Resources**
Functional Rights tab	
Functional rights (FuncRights)	Add the functional right **CP1-Human Resources**

c. Close all screens.

7. Create employees **CP1-Smith** and **CP1-Williams**

   These employees will access the Asset Manager database.

   a. Display the employees (**Organization management/ Organization/ Employees and departments** navigation menu).

   b. Create the following records by populating the fields and links as shown in the following tables:

Field or link	Value
Name	CP1-Smith
Title	Human Resources Advisor
Profile tab	
Access type	Floating
Profile	CP1-Human Resources Advisor
User name	smith
Password	Leave this field empty.

Field or link	Value
Name	CP1-Williams
Title	Human Resources Manager
Profile tab	

Field or link	Value
Access type	Floating
Profile	CP1-Human Resources Manager
User name	williams
Password	Leave this field empty.

c. Close all screens.

8. Look at the result for the user **Admin**;

   a. Display the navigation bar (**Window/ Display navigation bar** menu).

   b. Display the employees associated with the functional domain **CP1-Human Resources** (**CP1-Human Resources/ CP1_amEmplDept**) link on the navigation bar).

   c. Note that the full screen is displayed.

   d. This is because the user **Admin** is associated with the screen sets specified by the **Screen sets/ Default order** database option.
   In our example, the screen set order to use is **CP1-Complete,CP1-Simplified,Custom,Full,Simple**.

9. Look at the result for the user **Cp1 - Smith**

   a. Connect to the demonstration database as **Cp1 - Smith** (**File/ Connect to database** menu).

Field	Value
Login	smith
Password	The password is empty

   b. Display the employees associated with the functional domain **CP1-Human Resources** (**CP1-Human Resources/ CP1_amEmplDept**) link on the navigation bar).

   c. Note that the full screen is displayed.

   d. This is because the user **Cp1 - Smith** is associated with the profile **CP1 - Human Resources Advisor** and for this profile the value of the **Screen sets** (ScreenSets) field is **CP1-Complete, CP1-Simplified**.

10. Look at the result for the user **Cp1 - Williams**

Field	Value
Login	williams

Field	Value
Password	The password is empty

a. Connect to the demonstration database as **Cp1 - Williams** (**File/ Connect to database** menu).

b. Display the employees associated with the functional domain **CP1-Human Resources** (**CP1-Human Resources/ CP1_amEmplDept**) link on the navigation bar).

c. Note that the simplified screen is displayed.

d. This is because the user **Cp1 - Williams** is associated with the profile **CP1 - Human Resources Manager** and for this profile the value of the **Screen sets** (ScreenSets) field is **CP1-Simplified, CP1-Complete**.

Customization prepared in a test database

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

This section explains how to migrate structural changes from the test database to the working database (of the same version of Asset Manager).

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 HPE Connect-It.

This section includes:

Copying your working database 69

Propagating customizations from a test database to a production database 71
Exporting the database structure 72
Importing the new database structure 73
Migrate the data using HPE Connect-It 74
Look out for software counters 74

Copying your working database

Caution: 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 DBMS.

   See **Administration** guide, chapter **Creating, modifying and deleting an Asset Manager database**, section **Creating an empty shell with the DBMS**.

2. Create a DBMS login.

   See **Administration** guide, chapter **Creating, modifying and deleting an Asset Manager database**, section **Creating an empty shell with the DBMS**/ **Creating a DBMS login**.

3. Create a connection to the empty shell.

   See **Administration** guide, chapter **Creating, modifying and deleting an Asset Manager database**, section **Creating a connection with Asset Manager**.

4. Start Asset Manager Application Designer.
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).

See **Administration** guide, chapter **Creating, modifying and deleting an Asset Manager database**, section **Changing your DBMS**.

Note: When you use **Copy database to empty database** to copy a database, you can copy the

following extended database structure and data:

- Functional domains
- Profiles with access restriction and access rights
- Calculated fields
- Workflows
- Wizards and scripts
- Features

To do this, you need to enable an advanced option named **UseAdvancedDump**. To enable this option, on the **Edit** menu, hold the Control button and click **Options**, expand the **Advanced mode** node, and then set **UseAdvancedDump** to **Yes** .

After you enable the **UseAdvancedDump** option, when you run the **Copy database to empty database** command, a wizard pops up and allows you to select the specific tables to be copied. Notice that in order to enable the Admin user to log on to Asset Manager, the following tables must be included in the list.

- amOption
- amDateAlarm
- amItemizedList
- amCurrency
- amCurRate
- amFeatureClass
- amFeature
- amUnit
- amRelCalcFeat
- amFeatParam
- amFeatMemo
- amCalcField
- amCFAql
- amCFScript
- amTaxFormula

- amWfActivity
- amWfOrgRole
- amWfScheme
- amWfEvent
- amWkEvtScript
- amWfTransition
- amReconcProposal
- amAction
- amFuncDomain
- amReport
- amCbkRule
- amCbkSystem
- amViewDef
- amFieldAdjustTempl
- amSysConfig
- amEmplDept
- amLoginSlot
- amImage
- amMail
- SysBlob
- amQuery
- amActionMemo

Propagating customizations from a test database to a production database

There are two ways to customize the production database:

- By modifying the **gbbase*.*** files describing the structure of the database and importing those modifications into the production database.
- By modifying an empty test database and transfering 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 Asset Manager Application Designer

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 **gbbase*.*** 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 that you export the structures of both databases, compare them and integrate them using an integration program or a HPE Connect-It scenario.

Exporting the database structure

Asset Manager Application Designer 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 Asset Manager Application Designer.

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

> **Caution:** 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 Asset Manager Application Designer.
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.

Migrate the data using HPE Connect-It

Launch the HPE 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, Asset Manager 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 idenfier **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

> **Caution:** Implementing the process described in this section requires attention to detail and knowledge in the following areas:
>
> - Structure of the Asset Manager database
> - Exporting data
> - Importing data

- Database administration
- Administering Asset Manager

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

This section includes:

Explanation of the problem 75
Possible types of customizations 75
Recommended methodology 77
Global process 77
References: Customizable tables 91

Explanation of the problem

This section explains how to customize the Asset Manager 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 (named 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 Asset Manager 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, and so on)
- Fields (label, description, size, and so on)
- Links (label, description, default value, and so on)
- Indexes (label, description, list of fields, and so on)
- Pages (label, description, content, and so on)
- Screens (label, description, buttons, and so on)
- 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
- Views
- User rights and profiles, access restrictions
- Functional rights
- Database options and tips of the day
- Configurations and preferences
- Workflows and deployment servers

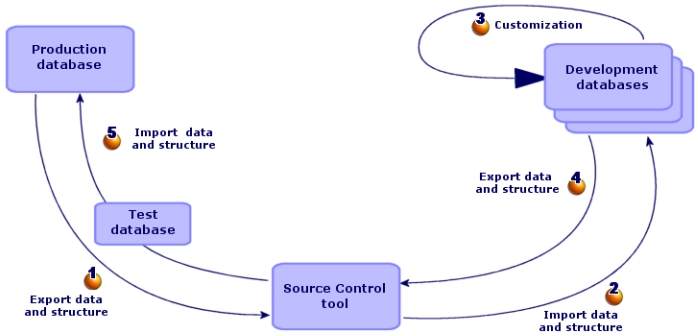
> **Note:** The choice of data to be customized depends on your needs.
>
> Once you have made your choice, make sure the tables corresponding to this data cannot be be modified by the Asset Manager users, exceptions permitted.

> 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.

Recommended methodology

Global process

Customization by several operators - simplified process



To customize Asset Manager as a team:

- Exporting data and structure of the production database
  a. Exporting the structure of the production database
  b. Exporting the data to be customized from the production database
  c. Storing the result of the export in a source control tool
  d. Creating test data

> **Note:** This series of steps is only performed once at the initial implementation stage of the project.

- Importing the data and structure of development databases
  a. Synchronizing files from the source control on the development computers
  b. Check out the files from the source control
  c. Creating development databases and importing data to customize and test data

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

- Customizing the development databases
  a. Customizing the structure of the development databases
  b. Customizing the data in the development databases
- 4 Exporting the data and structure of development databases
  a. Exporting the data of the customized development databases
  b. Exporting the customized data from the development databases
  c. Updating the files in the source control tool
- 5 Importing data and structure into the production database
  a. Creating a test database and importing the merged customized data
  b. Testing the test database
  c. Applying the customized structure to the production database
  d. Importing the customized data in the production database

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 Asset Manager Application Designer
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 guest users (not modifiable)
- **<Connection name>.wiz**: Functions library for wizards
- **<Connection name>.xml**: Structure of the database
- **<Connection name>_xx.str**: Contexual help on fields and links
- **<Connection name>_xx.xml**: Database dictionary
- **systemlstdet_xx.cfg**: Description of the screens
- **systemlstpage_xx.cfg**: Description of the pages
- **dawkfl.dsd**: Definition of the automation workflow (not modifiable)
- **tips.str**: Tips of the day

> **Note: xx** corresponds to the language version of **Asset Manager** (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 Asset Manager Export Tool
   See **Advanced use** guide, chapter **Exporting data and creating SQL views**
3. Connect to the production database
4. Select the **Edit/ Options** menu.
5. Expand 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)
9. Save this script under the name of your choice (**File/ Save script** menu).
   > **Note:** 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: See Database customization - Tables to take into consideration
11. For each query:
    a. Name the query (**Name** field)
       > **Tip:** We recommend using the SQL name of the exported table as the name of the query.
       >
       > Example: **amAction**
    b. Define the full path of the file that will be used to store the exported data (**File** field)
       > **Tip:** We recommend using the SQL name of the exported table as the file name.
       >
       > Example: **amAction.txt**
    c. Click the magnifier next to the **Query** field.
    d. Add the following elements from the list on the left to the list on the right:
       - All fields, except:

- **Modified on** (dtLastModif)
- **Hierarchic level** (sLvl)

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.
  See Database customization - Tables to take into consideration
  Do not use a primary key.

e. Enable sorting on the fields making up the reconciliation key (red arrow with a digit).

> **Caution:** This is required so that the exported files can be managed 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.

> **Note:** 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.

f. Confirm your choices by clicking **OK**

g. 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:** Asset Manager Export Tool uses the values of the **keyvalue** fields for certain files created during the export.

h. 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, **lIconId** 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 roll back changes if necessary, or compare changes between different versions.
- Merge 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.

> **Note:** 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 customizing the database, you may need to use test data with the development database. However it is probably not useful to store the test data under source control.

For example, when creating an action, an author needs to be assigned. 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 identify what they are going to modify and check out the corresponding files from the source control tool.

To check out the files from the source control tool:
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 DBMS

   See **Administration** guide, chapter **Creating, modifying and deleting an Asset Manager database**, section **Creating an empty shell with the DBMS**.

2. Create a DBMS login

   See **Administration** guide, chapter **Creating, modifying and deleting an Asset Manager database**, section **Creating an empty shell with the DBMS/ Creating a DBMS login**.

3. Create a connection to the empty shell

   See **Administration** guide, chapter **Creating, modifying and deleting an Asset Manager database**, section **Creating a connection with Asset Manager**.

4. Install required License Keys on the machine's instance of HP AutoPass License Management Tool.

   See **Administration** guide, chapter **Installing License Keys**.

   > **Tip:** HP AutoPass License Management Tool is automaticaly installed with any Asset Manager 9.61 component that you install.
   >
   > Installing License Keys is a manual process, though.

5. Start Asset Manager Application Designer
6. 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.
7. Start the database creation wizard (**Action/ Create database** menu).
8. 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 Asset Manager.
Creation	Create the database
Use avanced 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	Password to associate with the administrator login. **Note:** The Asset Manager database adminstrator is the record in the **Employees and departments** (amEmplDept) table for which the **Name** (Name) field is set to **Admin**. The database connection login in stored in the **User name** (UserLogin) field. The administration name is **Admin**.
Confirm password	Confirm the value entered in the **Password** field.

Create system data page:

Fields	Value
Use time zones	Select this option in order to be able to manage the time-zone differences between the database server, the date field and the Asset Manager clients. See **Advanced use** guide, chapter **Time zones**.
Server	Select the time zone of the Asset Manager database server. This is the time zone defined, for example, at the level of the Windows Control Panel of the server.
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.

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.

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

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

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

   > **Note:** Save regularly any modifications you make to the script.

8. Configure the script so that it imports the exported data files.

   See **Administration** guide, chapter **Import**.

9. For each file to be imported (1 for each file exported; see Database customization - Tables to take into consideration):
   a. Add the file (**File/ Add file**)
   b. 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
   c. Select the target table: Main table matching the imported file
   d. Map the file to the table (**Edit/ Map**)

e. Select the mapping **(file, target table)**

f. 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).

g. Define the keys recommended in the table Database customization - Tables to take into consideration as the reconciliation keys for the main table and linked tables

> **Caution:** 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 (**Import** button)
13. Make sure that the result of the import is consistent.

> **Note:** 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 Asset Manager Application Designer
2. Connect to your intance 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 same time.

To customize the data in a development database:

1. Start Asset Manager
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 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 greate, 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 Asset Manager Application Designer
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 sychronized the structure files

Exporting the customized data from the development databases

To export the data of a development database:

1. Start Asset Manager Export Tool
2. Connect to your instance of the development database
3. For each script corresponding to data to be exported:
   a. Open the script (**File/ Open script**)
   b. Execute the script (**Actions/ Execute the script**)

   See Exporting the data to be customized from the production database

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.

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.

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.

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.

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 Asset Manager Application Designer
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.

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.

Database customization - Tables to take into consideration

Table	Recommended key	Useful 1 links
Images		
Images (amImage)	Name (Name)	
Calculated fields		
Calculated fields (amCalcField)	SQL name (SQLName) + Table (TableName)	
Scripts		
Scripts (amScriptLibrary)	SQL name (SQLName)	
Units		
Units (amUnit)	Symbol (Symbol)	
Functional domains		
Functional domains (amFuncDomain)	SQL name (SQLName)	Parent domain (Parent.Code)
Queries		
Queries (amQuery)	SQL name (SQLName)	Author.IDNo (Author.IDNo)
		Domain.SQL name (Domain.SQLName)
Views		
View (amViewDef)	SQL name (SQLName)	Author.IDNo (Author.IDNo)
		Domain.SQL name (Domain.SQLName)
Options of the application		
Applications options (amOption)	• Section (OptSection) • Entry (OptEntry)	Employee.ID # (EmplDept.IDNo)
Configurations and preferences		
Configurations and preferences	• Entry (CfgEntry) • Section (CfgSection)	Employee.ID # (EmplDept.IDNo)

Database customization - Tables to take into consideration, continued

Table	Recommended key	Useful 1 links
(amSysConfig)	• Employee (EmplDept.IDNo)	
Reports		
Reports (amReport)	SQL name (SQLName)	Author.IDNo (Author.IDNo)
		Domain.SQL name (Domain.SQLName)
Itemized lists		
Itemized lists (amItemizedList)	Identifier (Identifier)	
Itemized list value (amItemListVal)	• Value (Value) • Itemized list.Identifier (ItemizedList.Identifier)	Itemized list.Identifier (ItemizedList.Identifier)
Features		
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)
Features (amFeature)	Export the records a second time with the **SQL name** (SQLName) and **Source features.SQL name** (SrcFeatures.SQLName) only **Note:** 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. Asset Manager Export Tool generates one line per link.	
Feature parameters (amFeatParam)	• Table (TableName) • Feature.SQL name (Feature.SQLName)	Help on this parameter.Description (ShortHelp.Comment)
		Help on this parameter.Example (ShortHelp.Example)
		Help on this parameter.Precautions (ShortHelp.Warning)

Database customization - Tables to take into consideration, continued

Table	Recommended key	Useful 1 links
		Feature.SQL name (Feature.SQLName)
Actions		
Actions (amAction)	SQL name (SQLName)	Author.IDNo (Author.IDNo)
		Deployment workflow.Reference (DaWfScheme.Ref)
		Domain.SQL name (Domain.SQLName)
		Action to execute.SQL name (ExecAction.SQLName)
		Report.SQL name (Report.SQLName)
User profiles		
User profiles (amProfile)	SQL name (SQLName)	Defaut action.SQL name (DefLoginAction.SQLName)
User rights (amUserRight)	SQL name (SQLName)	
Access restrictions (amAccessRestr)	SQL name (SQLName)	
Functional rights (amFuncRight)	SQL name (SQLName)	
User rights (amUserRight)	Export the records a second time with the **SQL name** (SQLName) and **Profiles.SQL name** (Profiles.SQLName) only **Note:** This second import enables you to create the links between user profiles and user rights during the import phase. Asset Manager Export Tool generates one line per link.	
Access restrictions (amAccessRestr)	Export the records a second time with the **SQL name** (SQLName) and **Profiles.SQL name** (Profiles.SQLName) only **Note:** This second import enables you to create the links between user profiles and access restrictions during the import phase. Asset Manager Export Tool generates one line per link.	

Database customization - Tables to take into consideration, continued

Table	Recommended key	Useful 1 links
Functional rights (amFuncRight)	Export the records a second time with the **SQL name** (SQLName) and **Profiles.SQL name** (Profiles.SQLName) only **Note:** This second import enables you to create the links between user profiles and functional rights during the import phase. Asset Manager Export Tool generates one line per link.	
Workflow schemes		
Employee groups (amEmplGroup)	Barcode (BarCode)	Supervisor.IDNo (Supervisor.IDNo)
		Sub-group of.Barcode (Parent.BarCode)
Employee groups (amEmplGroup)	Export the records a second time with the **Barcode** (BarCode) and **Members.IDNo** (Members.IDNo) only **Note:** This second import enables you to create the links between workflow groups and members during the import phase. Asset Manager Export Tool exports one line per line between a group and a member.	
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)	• Workflow.Reference (WfScheme.Ref) • Type (seType)	Workflow.Reference (WfScheme.Ref)
	Note: In this first file, add a WHERE clause to filter the START activities (seType = 0).	

Database customization - Tables to take into consideration, continued

Table	Recommended key	Useful 1 links
Workflow activities (amWfActivity)	• Workflow.Reference (WfScheme.Ref) • Reference (Ref)	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)	• SQL name (SQLName) • Activity.Reference (WfActivity.Ref) • Activity.Workflow.Reference (WfActivity.WfScheme.Ref)	Activity.Reference (WfActivity.Ref)
		Activity.Workflow.Reference (WfActivity.WfScheme.Ref)
Workflow events (amWfEvent)	• Workflow.Reference (WfScheme.Ref) • Source activity.Reference (SrcActivity.Ref) • Reference (Ref)	Source activity.Reference (SrcActivity.Ref)
		Workflow.Reference (WfScheme.Ref)
Workflow transitions (amWfTransition)	• Reference (Ref) • Source event.Reference (SrcEvent.Ref) • Source event.Source activity.Reference (SrcEvent.SrcActivity.Ref) • Workflow.Reference (WfScheme.Ref)	Source event.Reference (SrcEvent.Ref)
		Source event.Source activity.Reference (SrcEvent.SrcActivity.Ref)
		Target activity.Reference (TargetActiv.Ref)
		Workflow.Reference (WfScheme.Ref)
System activities (amWfSysActiv)	• Workflow.Reference (WfScheme.Ref) • Transition.Reference (WfTransition.Ref)	Transition.Reference (WfTransition.Ref)

Database customization - Tables to take into consideration, continued

Table	Recommended key	Useful 1 links
	• Transition.Source event.Reference (WfTransition.SrcEvent.Ref) • Transition.Source event.Source activity.Reference (WfTransition.SrcEvent.SrcActivity.Ref)	
		Transition.Source event.Reference (WfTransition.SrcEvent.Ref)
		Transition.Source event.Source activity.Reference (WfTransition.SrcEvent.SrcActivity.Ref)
		Workflow.Reference (WfScheme.Ref)
	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 **Workflow.Reference** (WfScheme.Ref), **Reference** (Ref) and **Actions.SQL name** (Actions.SQL name) only **Note:** This second import enables you to create the links between workflow groups and actions during the import phase. Asset Manager Export Tool generates one line per link.	
Workflow alarms (amWfActivAlarm)	• Reference (Ref) • Activity.Reference (WfActivity.Ref) Activity.Workflow.Reference (WfActivity.WfScheme.Ref)	Action.SQL name (Action.SQLName)
		Activity.Reference (WfActivity.Ref)
		Activity.Workflow.Reference (WfActivity.WfScheme.Ref)
Calendars		
Time zones (amTimeZone)	Code (Code)	
Calendar of business days (amWorkCalendar)	SQL name (SQLName)	Time zone.Code (TimeZone.Code)
Currencies		
Currencies (amCurrency)	Name (Name)	
Exchange rates (amCurRate)	• Source currency.Name (SrcCurrency.Name)	Source currency.Name (SrcCurrency.Name)

Database customization - Tables to take into consideration, continued

Table	Recommended key	Useful 1 links
	• Target currency.Name (TargetCurrency.Name) Date of rate (dtStart)	
		Target currency.Name (TargetCurrency.Name)
HelpDesk		
Severity (amSeverity)	Name (Name)	
Escalation schemes (amEscalScheme)	Barcode (BarCode)	Assignment.SQL name (AssignAct.SQLName)
		Ticket supervisor change.SQL name (AssignChangeAct.SQLName)
		Calendar.SQL name (Calendar.SQLName)
		Close.SQL name (CloseAct.SQLName)
		Group change.SQL name (GrpChangeAct.SQLName)
		Deletion of group.SQL name (GrpUnassignAct.SQLName)
		Opening.SQL name (OpenAct.SQLName)
		In progress.SQL name (ResolAct.SQLName)
		Deletion of ticket supervisor.SQL name (UnassignAct.SQLName)
		Closed and verified.SQL name (VerifiedAct.SQLName)
Escalation schemes (amEscalScheme)	Export the records a second time with the **Barcode** (BarCode) and **Locations.Barcode** (Locations.BarCode) only **Note:** This second import enables you to create the links between escalation schemes and locations during the import phase. Asset Manager Export Tool generates one line per link.	
Escalation schemes (amEscalSchem	Export the records a third time with the **Barcode** (BarCode) and **Severities.Name** (Severities.Name) only	

Database customization - Tables to take into consideration, continued

Table	Recommended key	Useful 1 links
e)	**Note:** This second import enables you to create the links between escalation schemes and severities during the import phase. Asset Manager Export Tool generates one line per link.	
Escalation levels (amEscSchLevel)	• Name (Name) • Escalation.Barcode (EscalScheme.BarCode)	Action.SQL name (Action.SQLName)
		Escalation.Barcode (EscalScheme.BarCode)

Remarks: Choice of reconciliation keys

The choice of reconciliation 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; Asset Manager 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:
  a. 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**.
  b. 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**).
  c. 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**.
  d. When it is imported into the production database, the action **A1** is identifed in the production database by its SQL name (**SQL1**), which is used as a reconciliation key. The record describing the action **A1** is updated.

Database options

Certain options specifically concern a database.

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

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

This section includes:

To configure the database options 100
Database option configuration window 101

To configure the database options

1. Launch Asset Manager.

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

2. 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 Asset Manager 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.

Database options - Description of the available options

Section	Name of the option	Description	Example of values
Procurement	Create expense lines for consumables as soon as received	• **Yes**: 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 (that is, 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 **consumption** aspect. • **No**: No expense line is created when a consumable is created upon its receipt. An expense line is created when the consumable is used (that is, when the consumable is linked to the parent asset that consumes it).	Yes
Procurement	Let Asset Manager Automated Process Manager create the items received in the portfolio	• **Yes**: Receiving an item from a purchase order will create a record in the **Receiving slips** table (amReceipt). However, the received items are not created in their respective tables (assets, computers, portfolio items, and so on). Temporary records are created in the **Items received** table (amItemsReceived) and processed by the **Create assets, consumables, and so on corresponding to items received** module in Asset Manager Automated Process Manager, which takes care of creating records in the final tables. Purpose of this method: Giving this task to Asset Manager Automated Process Manager rather Asset Manager enables increased	Yes

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
		performance for those client computers used to receive purchase orders. • **No**: Receiving an item from a purchase order will create a record in the **Receiving slips** table (amReceipt). The corresponding records are immediately created in their respective tables (assets, computers, portfolio items, and so on).	
Import	Disable history during import	• **Yes**: No record is created in the **History** table (amHistory) for modifications brought about by importing the data. • **No**: 1 record is created in the **History** 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.	Yes
Wizards	BASIC functions	This option stores Basic functions that can be called in **Wizard** type actions.	
Tips of the day	Text of tips	List of tips of the day displayed when Asset Manager is launched.	TIP2, "Shift+F9 starts the wizard debugger."
LDAP authentication	Use an LDAP server for authentication	• **Yes**: The password entered when connecting to the database is looked up in an external LDAP directory. • **No**: The password entered when connecting to the database is looked up in the **Password** field (LoginPassword) of the **Employees and Departments** table (amEmplDept).	Yes
LDAP authentication	Use LDAP authentication first, if it fails, use AM authentication	• **Yes**: The password entered when connecting to the database is looked up in an external LDAP directory first. If the authentication fails, the Asset Manager authentication is used. • **No**: The Asset Manager authentication is not used even if the LDAP authentication fails. This option is only taken into account if the **Use an LDAP server for authentication** option is	Yes

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
		set to **Yes**.	
LDAP authentication	Authorize anonymous LDAP connections (empty login without password)	• **Yes**: Anonymous LDAP connections are authorized. • **No**: Anonymous LDAP connection is disabled and the LDAP server checks login and password validity. This option is only taken into account if the **Use an LDAP server for authentication** option is set to **Yes**.	Empty
LDAP authentication	Use a secure connection (SSL) to the LDAP server	• **Yes**: Authentication is performed via the LDAP certificate directory (**LDAPCertDir** or **LDAPCertfile**). • **No**: The password entered when connecting to the database is looked up in an external LDAP directory. This option is only taken into account if the **Use an LDAP server for authentication** option is set to **Yes**.	Yes
LDAP authentication	LDAP server name	Name of the server exposing the directory in which the Asset Manager user logins and passwords are stored. This option is only taken into account if the **Use an LDAP server for authentication** option is set to **Yes**.	MyLDAPServer
LDAP authentication	LDAP server port	LDAP server port used to listen for requests concerning the directory. This option is only taken into account if the **Use an LDAP server for authentication** option is set to **Yes**.	389
LDAP authentication	SQL name of the 'amEmplDept' table in which the LDAP directory login is stored.	By default, this option supposes that you are using the same login in the Asset Manager database and in the LDAP directory: This login is then stored in the **Login** field (UserLogin) of the **Employees** table (amEmplDept). You may choose to differentiate between the LDAP directory login and that of Asset Manager. You must select a field to store the LDAP	Field1

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
		directory login and indicate its SQL name using this option. This option is only taken into account if the **Use an LDAP server for authentication** option is set to **Yes**.	
LDAP authentication	Authenticate Asset Manager before user authentication	If this option is set to **Yes**, the Asset Manager client will be authenticated before the user authentication takes place. By default, this option is set to **No**.	Yes
LDAP authentication	LDAP DN of Asset Manager	The distinguished name of the LDAP server that is used to authenticate the Asset Manager client.	
LDAP authentication	LDAP password of Asset Manager	The password of the LDAP server that is used to authenticate the Asset Manager client.	
LDAP authentication	The minimum level of the SSL/TLS protocol. (0=SSLv2, 1=SSLv3, 2=TLS1.0, 3=TLS1.1, 4=TLS1.2)	The minimum level of the SSL/TLS protocol. By default, this option is set to 4.	4
LDAP authentication	LDAP cipher suite	This option specifies the cipher suite list of SSL. For more information about the format of the cipher list, see https://www.openssl.org/docs/apps/ciphers.html.	
Authorization	Asset Manager Automated Process Manager signature	Used by Asset Manager to verify that Asset Manager Automated Process Manager regularly connects to the database. This option cannot be modified in this window.	Updated by Asset Manager Automated Process Manager.
Features	Disable recalculation of	• **Yes**: The values of consolidated features are not be recalculated when the values of the features making them up change.	Yes

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
	consolidated features	• **Yes**: The values of consolidated features are recalculated when the value of at least one of the features making them up changes.	
Access control	Discard current changes after automatic disconnection	Ensures that changes are only saved intentionally.	Yes
Access control	Enable automatic disconnection of named users	Ensures that user sessions do not remain open indefinitely.	Yes
Access control	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 **Enable automatic disconnection** option is set to **Yes**.	1800
Administration user authorization	Enable the administration user to access other users' message	Selecting **YES** will enable users with administration rights to be able to read the messages of others'. By default, the option is set to **NO**.	Yes
Agent	Select YES to stop checking the change of computer type of nature	Selecting **YES** will enable the user to change the computer type nature in the model without any agent check. For example, you can change it from **Computer** to **Virtual machine**. By default, the option is set to **NO**.	YES
Documents	Maximum size of documents that can be inserted into	This is maximum size of files that can be inserted in the **Documents** table (amDocument).	5 242 880

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
	the database (bytes)		
System data	Build	Build number of the the **System data** imported into the database. The **system data** 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 (minutes)	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.	-120 (time zone GMT-120)
Time zone	Server: Time difference in relation to the reference GMT time zone (minutes)	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.	240 (time zone GMT+240)
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
Homepage security	Deactivate the white list, which contains the functions that are allowed to be used in AQL script of homepage widget	Setting this option is set to "No" turns on the function white list for AQL-type homepage widgets. Only the functions listed in **The white list of functions that are allowed to be used in AQL script of homepage widget, separated by comma, case-insensitive** option can be used in AQL-type homepage widgets. By default, this option is set to "No".	No

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
Homepage security	The white list of functions that are allowed to be used in AQL script of homepage widget, separated by comma, case-insensitive	The white list of the functions that are allowed to be used in AQL-type homepage widget. This options takes effect only when the **Deactivate the white list, which contains the functions that are allowed to be used in AQL script of homepage widget** option is set to "No".	HOST_ ID,HOST_ NAME,APP_ NAME
Screen sets	Default order	This option defines the list of screen sets to use starting with the highest priority first (example: **Full, Simple**). This option is used when the following user types request a table be displayed: • Users associated with a profile whose **Screen sets** (ScreenSets) field is empty • Users associated with a profile whose **Administration rights** (bAdminRight) option is selected When these users request a table be displayed, Asset Manager takes the identifiers of the screen sets in the order specified by the option and searches the table to see if it contains a screen associated with this screen set. When a screen is found, Asset Manager uses it to display the table. This default behavior can be superseded by the functional rights associated with the profile: Access to a given screen can be denied by the functional rights. For example: Employees in the **Employees and departments** (amEmplDept) table can be displayed using two screens: • A full screen: SQL name: **amEmplDept** Screen set: **Full**	Full,Simple

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
		• A simple screen: SQL name: **amEmplDept** Screen set: **Simple** Suppose that user **Hartke** is associated with a user profile that allows him to view the **Employees and departments** table and that the **Screen sets** (ScreenSets) field is empty. If the **Default order** database option is **Full,Simple**, then the **Organization/ Employees and departments** link on the navigation bar will display the **amEmplDept** screen associated with the **Full** screen set.	
Leasing	Authorize automatic asset LRF calculation	• **Yes**: The **LRF** field (pLRF) of assets is automatically calculated by Asset Manager depending on the values available. • **No**: 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.	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
Full-text	Enable full-	By default, simple filters enable you to use the	Yes

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
search	text search	**LIKE** operator. This operator searches a string within the values of a field. This search is performed directly by Asset Manager. Certain DBMSs offer equivalent functionality with greater efficiency. When this is the case, this function must be specifically enabled at the DBMS level in order for it to be available. This search function then uses the **CONTAINS** operator instead of the **LIKE** operator. The **Enable full-text search** option enables the **CONTAINS** operator in the simple filter editor.	
Query generation options	Applied to get query generating	A "get query" is used to retrieve the fields of one record in the database. • **Yes**: Constant format uses the text format for get query. • **No**: Constant format does not use the text format for get query. **Note:** This option can also be set by modifying the **ConstAsTextForGet** parameter in the `amdb.ini` file. If it is already set in the `amdb.ini` file, the database option does not take effect.	Yes
Query generation options	Applied to select query generating	A "select query" is used to retrieve multiple records in the result set. • **Yes**: Constant format uses the text format for select query. • **No**: Constant format does not use the text format for select query. **Note:** This option can also be set by modifying the **ConstAsTextForSelect** parameter in the `amdb.ini` file. If it is already set in the `amdb.ini` file, the	Yes

Database options - Description of the available options, continued

Section	Name of the option	Description	Example of values
		database option does not take effect.	
Unicode Query	Use Unicode string constants in Microsoft SQL Server queries	If the **Applied to get query generating** or the **Applied to select query generating** option is set to **Yes**, this database option takes effect. • **Yes**: The constant text in the SQL statement is in unicode. • **No**: The constant text in the SQL statement is in ANSI. **Note:** This option only works for SQL Server.	Yes
DB engine logs control	Show/hide the low level DB logs on the message box	If this option is set to **Yes**, message boxes (for example, an error message box) do not show low-level database operations to end users. On the opposite, when this option is set to **No**, message boxes reveal low-level database operations to all users. By default, this option is set to **Yes**.	Yes

Customizing Screens

This chapter introduces how to customize Asset Manager screens.

Customizing screens 111
Definition of a screen 111
Differences between screens and views 112
To create a screen from scratch in Asset Manager Application Designer 113
To create a screen from a view in the Windows client 113
Managing filters in screens 114
To display a screen 115
To modify a screen 116
To delete a screen 116

Customizing screens

This section explains how to use screens.

Definition of a screen 111
Differences between screens and views 112
To create a screen from scratch in Asset Manager Application Designer 113
To create a screen from a view in the Windows client 113
Managing filters in screens 114
To display a screen 115
To modify a screen 116
To delete a screen 116

Definition of a screen

Screens let you store a specific state of the database that you can easily view on a regular basis.

They can be accessed via the navigation bar: See To display a screen.

For more information about screens, read paragraph Creating a screen.

Views are similar, but are only available in the Windows client, whereas screens are available in both the Windows and Web clients.

Screens can be viewed by all users.

Views can either be viewed only by the view's owner (or users with administration rights) or by all users. This is defined by the **Not shared** (amViewDef:bPrivate) field.

For more information about **Views**, read chapter Using Views.

> **Tip:** Screens can only be created, modified or deleted from Asset Manager Application Designer.
>
> Once created, they can be displayed in both the Windows and Web clients.

Differences between screens and views

Screens have several advantages over views: they can be displayed in the Web client, and they are available to all users.

Differences between a view and a screen

	View	**Screen**
Available in the Windows client	Yes	Yes
Available in the Web client	No	Yes
Visible by all users	Defined by the **Not shared** (amViewDef:bPrivate) field	Yes
Can be viewed by users with administration rights	Yes	Yes
Storage	**Views** (amViewDef) table	Database structure
Creating	Windows client	Windows client or Asset Manager Application Designer
Modifying and deleting	Windows client	Asset Manager Application Designer

If you create a screen based on a view, (see To create a screen from a view in the Windows client), it will have the same properties as the view.

To create a screen from scratch in Asset Manager Application Designer

See Creating a screen

To create a screen from a view in the Windows client

1. Start the Windows client and connect to the database.
2. Display the list of views (**Tools/ Views/ Edit views...** menu).
3. Select the view that you want to share with other users or make available via the Web client.
4. Click **+Screen**.

   This starts the **Create screen using view as a template...** (sysCoreViewToScreen) wizard.
5. Complete the wizard, and make sure you specify a value for the **Functional domain** field in order for the screen to be accessible via a node under that domain on the navigation bar.
6. Exit the wizard.

> **Note:**
>
> - You should keep the view once the screen has been created in case you would like to make changes to the screen later on; it will be easier to start from the view rather than from the screen in Asset Manager Application Designer; furthermore, you cannot change the set of pages of a screen with Asset Manager Application Designer when the screen originates from a view.
> - If you have set filters for the view, the filters may be modified when you create the screen. For more information on how filters are managed when creating a screen, read section Transforming filters when creating the screen.

When a screen is created out of a view, the following limitations apply:

- The customization of tab lists are not saved for the following tabs: **History**, **Documents**, **Workflow**, **Features**, **Adjustments**.
- Simple and cascading filters are converted to a single query filter when you create a screen.

- Only simple **sorts** are kept for main lists when a screen is created.
- If you set a default filter on a tab list (collection) through Asset Manager Application Designer this is not applied by default in the Web client. It needs to be re-selected manually in the list of filters available for that collection.
- The following properties available in the Windows client views using the **Utilities/ Configure list** contextual menu are not supported in the Web client via screens:
  - **Columns and sort** tab:
    - Sorting on multiple columns (the first one is retained)
    - **Force indexes**
    - **View NULLs**
    - **Color**
  - **Filter (WHERE clause)** tab:
    - The whole **Where** clause is ignored for tab lists (but kept for main lists)
- Filters that refer to monetary fields will not work in the Web client.

Managing filters in screens

When you create a screen from a view, you define and associate filters.

This section explains how filters are managed when creating a screen.

Transforming filters when creating the screen

When you create a screen from a view, filters are transformed as indicated in the following table:

Filters used for the view or window	Result in the screen
Simple filter	This filter type is replaced with a query filter in which the operator is hard coded. This filter will be visible in the screen.
Several filters (any type)	A single query which concatenates each of the conditions from all of the filters is generated.
List filter (WHERE clause) with the	This filter, which is initially hidden (not visible by a standard user), is converted to a visible query filter and will be concatenated with any other

Filters used for the view or window	Result in the screen
icon	display filters.
List filter (WHERE clause) set via the icon for **tab lists**	This filter is dropped when the screen is created.

To modify the filter applied to the screen's main list

1. Start Asset Manager Application Designer.
2. Connect to the database.
3. Display the table of which the view is a part.
4. Display the screens (**View/ Screens** menu item).
5. Select the screen for which you wish to define filters.
6. Display the **List/Detail** tab.
7. Note the value of the **Query filter** field.

   This is the SQL name of the query that filters the screen's main list.
8. Exit Asset Manager Application Designer.
9. Start the Windows or Web client.
10. Connect to the database.
11. Display the queries (**Administration/ Queries** navigation menu).
12. Select the query using the SQL name that you noted from Asset Manager Application Designer.
13. Modify the query.
14. Save the modifications.

To display a screen

To display a screen: Navigate to the screen in the navigation bar (under the **Functional domain** that you specified when you created the screen).

> **Note:** If the screen has just been created, you will need to re-connect to the database to be able to see it.

To modify a screen

1. Start Asset Manager Application Designer and connect to the database.
2. Display the table underlying the view.
3. Display the screens for that table (**View/ Screens** menu item).
4. Select the screen that you want to modify.

   > **Note:** You cannot make changes to the set of pages.

5. Modify its properties.
6. Save the modifications made to the screen (**Modify** button).
7. Save the changes to the database (**File/ Update database structure...** menu).

To delete a screen

1. Start Asset Manager Application Designer.
2. Connect to the database.
3. Display the table of which the view is a part.
4. Display the screens (**View/ Screens** menu item).
5. Select the screen that you want to delete.

   > **Note:** You cannot delete System screens (screens for which the **Domain** is System).

6. Click **Delete**.
7. Save the changes to the database (**File/ Update database structure** menu).

> **Caution:** When deleting a screen using this method, if a query filter was associated with the view it will not be deleted from the queries table.

Customizing Windows Clients

This chapter introduces how to customize Asset Manager Windows clients.

Customizing a Windows client 117
Activating the modules 118
Customizing the toolbar 118
Customizing tabs 120
General options 121
User defined customizations 123
User defined colors in scripted queries for charted statistics 124
Using Views 125
Definition of a view 126
Creating a view 128
Displaying a view (Windows client) 129
Deleting a view 129
Modifying a view 130
Editing a view 130
Associating a view with a button in a screen 130

Customizing a Windows client

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

You will learn how to:

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

All users can customize Asset Manager 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.

This section includes:

Activating the modules 118
Customizing the toolbar 118
Customizing tabs 120
General options 121
User defined customizations 123
User defined colors in scripted queries for charted statistics 124

Activating the modules

Asset Manager 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 simply the user interface.

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

To enable or 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.

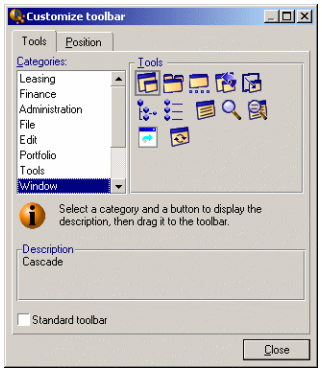
2. Click **OK** to validate your modifications.

Customizing the toolbar

This section explains how to customize the Asset Manager 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.

   Asset Manager 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.

   Asset Manager 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. Asset Manager 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. Asset Manager 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). Asset Manager 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. Asset Manager deletes the separator and resizes the toolbar if necessary.

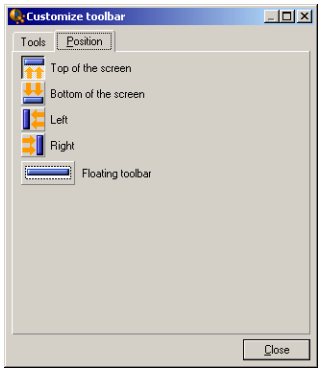
Displaying the default toolbar

Asset Manager 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.



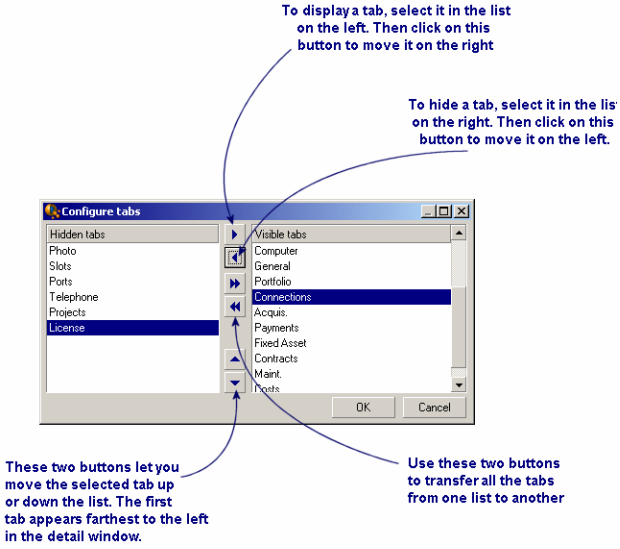
Click the button corresponding to the position in which you want to find your toolbar. Or, you can move the toolbar anywhere you like on the workspace by choosing **Floating** toolbar.

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.

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

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

> **Note:** Thes 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.

  > **Caution:** 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.

  > **Caution:** 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 database-dependent options.

  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 database-dependent or not:
>
> 1. Launch Asset Manager 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.

User defined customizations

Enforce default configuration

By default, Asset Manager stores the user preferences (also named 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, and so on), 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 Asset Manager. 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 [lEmplDeptId]=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**.

> **Note:** 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.

User defined colors in scripted queries for charted statistics

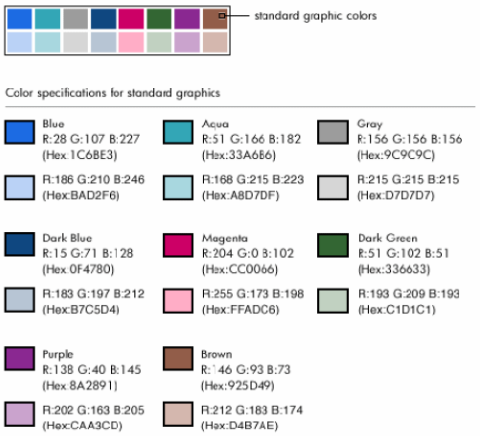
Users can specify a set of custom colors to be used by a scripted query which returns charted statistics results for a set of subgroups or series. (It only makes sense to define colors in queries which involve multiple subgroups.)

To define a color for a series, use the **color=** parameter with the subgroup definition statement. For example

```
  <subgroup Name="Germany" filtertable="amEmplDept" filter="Location.FullName like
'/Germany/%' and MrMrs='Mr'" value="9" color="#cc0066"/>
```

> **Note:** To ensure compatibility with Asset Manager Windows and Web clients, colors need to be specified with the hex equivalent of their RGB values, for example **#6f006f** for purple.

Standard graphic colors are represented with the following values (however you can specify any other color with the hex equivalent of its RGB value) :



Error handling

The following rules are used to resolve anomalies in color definitions:

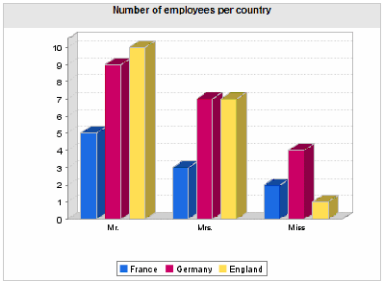
- If the **color=** parameter is undefined for one ore more subgroups in a query, default colors will be applied to those subgroups.
- If the **color=** parameter uses an unrecognized value, a default color will be applied.
- Colors can only be defined once per subgroup. (Any attempt to redefine them will be ignored.)

Practical Case

The following query defined below defines the customs colors blue, purple and yellow for three subgroups of the **Mr** group.

```
<group Name="Mr.">
<subgroup Name="France" filtertable="amEmplDept" filter="Location.FullName like
'/France/%' and MrMrs='Mr'" value="5" color="#1c6be3" />
<subgroup Name="Germany" filtertable="amEmplDept" filter="Location.FullName like
'/Germany/%' and MrMrs='Mr'" value="9" color="#cc0066"/>
<subgroup Name="England" filtertable="amEmplDept" filter="Location.FullName like
'/England/%' and MrMrs='Mr'" value="10" color="#ffde53"/>
</group>
```

The query results in the following chart:



To learn more about defining scripts to display statistics: see **Advanced use** guide, chapter **Statistics** , section **Examples of statistics/ Scripted statistics**.

Using Views

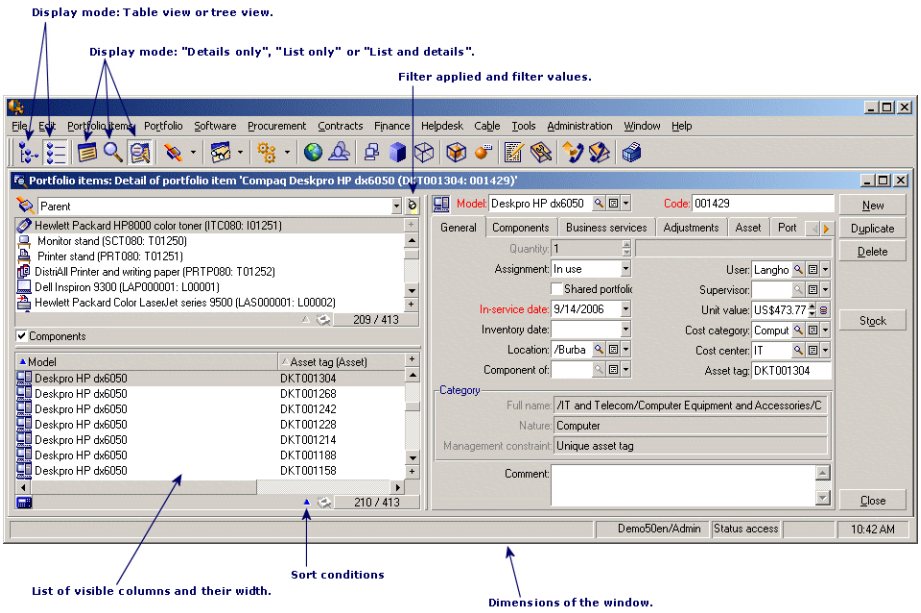
This section explains how to create, edit, display and delete views.

> **Tip:** Views are only available via the Windows client (not the Web client).

> To have an equivalent in the Web client:
>
> - Create a screen using Asset Manager Application Designer: See Creating a screen
> - Create a screen from a view in the Windows client: See To create a screen from a view in the Windows client

Use the **Tools/ Views** menu to create, edit, display and delete views.

Items stored in a view



This section includes:

Definition of a view 126
Creating a view 128
Displaying a view (Windows client) 129
Deleting a view 129
Modifying a view 130
Editing a view 130
Associating a view with a button in a screen 130

Definition of a view

A view lets you save and retrieve customized display parameters for a main list:

- The list display mode: Table view or Tree view.
- The screen display mode: Details only, List only, or List + Details.
- The position of the list with respect to the detail: horizontal or vertical.
- List sort conditions.
- Filter applied to the list, and the filter values.
- The list and the width of the columns visible in the list.
- The position of the displayed window.
- Dimensions of the displayed window.
- The list of tabs visible in the detail.
- Display or do not display records in the list when the value of a field used to sort a list is null.
- Use or do not use forced indexes when the list is sorted.
- Grouping by fields or links in the list.
- Display mode of tab lists: in list form or tab form.
- Color of lines in lists.

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 more than $5,000.
- List of all offices with a fax machine, displaying the office number, the fax model and the name of their users.

> **Note:** Views can either be viewed only by the view's owner (or users with administration rights) or by all users. This is defined by the **Not shared** (amViewDef:bPrivate) field.

To create a screen that can be displayed both in the Windows and in the Web client:

- Using Asset Manager Application Designer: See Creating a screen
- From a view in the Windows client: See To create a screen from a view in the Windows client

> **Note:** Views are stored in the **Views** table (amViewDef).

You can export the definitions of all the views by running the **Export all views definition to file** wizard. By default, this wizard is under **Administration** > **User actions**. Then, the definitions are stored as a .txt file.

> **Note:** This wizard can only be run by the following users:
>
> - The author of the wizard.
> - A user with the **Administration** user right.

Creating a view

1. Start the Asset Manager Windows client.

   > **Note:** Views cannot be created via the Web client, nor can they be viewed in the Web client (except if they are saved as screens).

2. To create a view, display the list exactly as you want it - these parameters will be saved for future retrieval.
3. Save this view via the **Tools/ Views/ Create view from the current window...** menu.

Any Asset Manager user with access to the **Views** (amViewDef) table can create a view.

Views can either be viewed only by the view's owner (or users with administration rights) or by all users. This is defined by the **Not shared** (amViewDef:bPrivate) field.

To make a view available in the Web client, you must create a screen equivalent to the view:

- Using Asset Manager Application Designer: See Creating a screen
- From a view in the Windows client: See To create a screen from a view in the Windows client

An Asset Manager administrator may display, modify and delete views made by all users, including the ones users decide not to make public.

> **Note:** When you create a view from a screen and the screen's column list contains columns that were not in the original one, if you want non-admin users to be able to see the view accurately, use one of the following methods.
>
> **Method 1**
>
> In Application Designer, add the additional columns to the **Other columns** box of the screen from which the view is generated.

Method 2

In the Windows client, open the user role record of the user who is supposed to see the view, check the **Authorize display of all fields and links in the lists** check box.

> **Note:** Users associated with a user role whose **Authorize display of all fields and links in the lists** (bFullListCfg) check box is selected are not limited in the selection of columns that they can add to a list.

Displaying a view (Windows client)

A view can be displayed in different ways.

Views can either be viewed only by the view's owner (or users with administration rights) or by all users. This is defined by the **Not shared** (amViewDef:bPrivate) field. This is true as long as the **Visible in the menus** option (bVisible) has been checked.

The view is displayed using all the saved parameters.

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

There are three ways to display a view in the Windows client:

- Select the view to display via the **Tools/ Views** menu.
- Select the desired view from the list that is displayed by clicking the button on the toolbar. This button displays the icon of the view that was last called.
- Select the desired view in the navigation bar's tree structure that corresponds to the **Domain** (Domain) field that you specified when you created the view.

Deleting a view

- 1st method:
  a. Start the Asset Manager Windows client.
  b. Display the view.
  c. Delete it using the **Tools/ View/ Drop current view** menu item.

- 2nd method:
  a. Start the Asset Manager Windows client.
  b. Select the **Tools/ Views/ Edit views...** menu item to display the table of views.
  c. Select the view you want to drop.
  d. Click **Delete**.

> **Note:** Views can neither be displayed nor deleted via the Web client.

Modifying a view

1. Start the Asset Manager Windows client.

   > **Note:** Views can neither be displayed nor modified via the Web client.

2. Display the view.
3. Modify your display parameters.
4. Save your changes using the **Tools/ Views/ Modify current view** menu item.

Editing a view

1. Start the Asset Manager Windows client.

   > **Note:** Views can neither be displayed nor edited via the Web client.

2. Display the views (**Tools/ Views/ Edit views...** menu).
3. Select the view to edit.
4. Modify the view's parameters.
5. Save the modifications (**Modify** button).

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 Buttons.
- This guide, section Creating action buttons.

Customizing Web Clients

This chapter introduces how to customize Asset Manager Web clients.

Modifying the Web client's default behavior 133
Disabling authentication when accessing the WSDL 134
Changing the text and the image of the banner 136
Reducing the time it takes to display screens for the first time 137
Defining the number of lines to export to Excel 139
Defining the number of lines to display in the lists 139
Modifying text read by screen reader software 140
Defining the number of records to display in the drop-down lists 141
Choosing the location of the workspace toolbar 141
Modifying how often the cache managing the workflow schemes and instances is refreshed 142
Refreshing a detail screen when the cursor leaves a field's or link's edit zone 143
Modifying the delay before a link's list of values is refreshed 144
Using other parameters to generate the Asset Manager Web Tier log file 144
Using another screen from the Business home pages table to display business home pages 145
Modifying the style sheet to use to display and print HTML pages from Asset Manager Web Tier 146
Modifying how often the currency cache is refreshed 146
Modifying how often the itemized lists cache is refreshed 147
Enabling the confirmation page before a script type action is executed 148
Modifying the maximum number of segments to display in report or statistics graphs 148
Modifying the maximum number of reports to display in dashboards 149
Modifying the maximum number of segments to display in graphs for business home pages 149
Modifying the maximum number of statistics to display in business home pages 150
Modifying the root used for URLs in Asset Manager Web Service 150
Modifying the root used for URLs in Asset Manager Web Service 151
Modifying the root used for URLs in Asset Manager Web Tier 152

Modifying the name of Asset Manager Web Service that is displayed in administration screens of your application server ... 152
Modifying the name of Asset Manager Web Tier that is displayed in administration screens of your application server ... 153
Creating a logon disclaimer for the Web client ... 154
User defined customizations ... 155
Homepage of the Web client ... 163
How to create a homepage widget ... 164
Personalizing the homepage ... 184
User profiles ... 184
How to create a homepage template ... 186
Importing/exporting homepage widgets and templates ... 186
Business home pages ... 189
What is a business home page? ... 190
How business home pages work ... 190
How to create or modify a business home page ... 191
Recommendations when creating business home pages ... 191
HTML syntax for useful links ... 191
Specifying default home pages and home pages based on user roles ... 193
Deleting a business home page ... 195

Modifying the Web client's default behavior

This section includes:

Disabling authentication when accessing the WSDL ... 134
Changing the text and the image of the banner ... 136
Reducing the time it takes to display screens for the first time ... 137
Defining the number of lines to export to Excel ... 139
Defining the number of lines to display in the lists ... 139
Modifying text read by screen reader software ... 140
Defining the number of records to display in the drop-down lists ... 141
Choosing the location of the workspace toolbar ... 141

Modifying how often the cache managing the workflow schemes and instances is refreshed 142
Refreshing a detail screen when the cursor leaves a field's or link's edit zone 143
Modifying the delay before a link's list of values is refreshed 144
Using other parameters to generate the Asset Manager Web Tier log file 144
Using another screen from the Business home pages table to display business home pages 145
Modifying the style sheet to use to display and print HTML pages from Asset Manager Web Tier 146
Modifying how often the currency cache is refreshed 146
Modifying how often the itemized lists cache is refreshed 147
Enabling the confirmation page before a script type action is executed 148
Modifying the maximum number of segments to display in report or statistics graphs 148
Modifying the maximum number of reports to display in dashboards 149
Modifying the maximum number of segments to display in graphs for business home pages 149
Modifying the maximum number of statistics to display in business home pages 150
Modifying the root used for URLs in Asset Manager Web Service 150
Modifying the root used for URLs in Asset Manager Web Service 151
Modifying the root used for URLs in Asset Manager Web Tier 152
Modifying the name of Asset Manager Web Service that is displayed in administration screens of your application server 152
Modifying the name of Asset Manager Web Tier that is displayed in administration screens of your application server 153
Creating a logon disclaimer for the Web client 154
User defined customizations 155

Disabling authentication when accessing the WSDL

During the initial connection to the WSDL, Internet Explorer displays an authentication window requesting an Asset Manager user name and password.

This reinforces security when accessing Asset Manager databases.

Authentication can be disabled.

This allows applications that do not support WSDL authentication to access the Asset Manager database.

> **Note:** Applications can only access the Asset Manager WSDL if the License Keys installed for Asset Manager include authorization for the Web services.
>
> Also, if authentication is enabled, users can only access the WSDL if their user profile provides them the functional right to access the Web services.

To disable authentication:

1. Decompress the **.war** file used to install Asset Manager Web Service in the **AssetManagerWebService.war_build** folder.
   It is assumed that you are using the **C:\Program Files\HPE\Asset Manager 9.61 xx\websvc\AssetManagerWebService.war** file.
2. Edit the **AssetManagerWebService.war_build\WEB-INF\spring\application-context.xml** file.
3. Remove the comment marks **<!--** and **-->** from the following section:

   ```
   <property name="convertUrlToLowerCaseBeforeComparison">
     <value>true</value>
   </property>
   ```

4. Locate the following section:

   ```
   <property name="publicResources">
     <list>
       <value>/index.jsp</value>
       <value>/images/**</value>
     </list>
   </property>
   ```

5. Add the following lines:

   ```
   <value>/**/*?wsdl*</value>
   <value>/**/*.xsd</value>
   ```

6. Save the **application-context.xml** file.
7. Open the **AssetManagerWebService.war_build** folder.
8. Select all files and folders.
9. Compress these files and folders in the **webapps\AssetManagerWebService.war_build\AssetManagerWebService.war** file.
10. Copy the file.
11. Paste it in the **webapps** folder (overwrite the previous file).
12. Delete the **webapps\AssetManagerWebService.war_build** folder.

13. Reinstall Asset Manager Web Service: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Changing the text and the image of the banner

Changing the image of the banner

The images correspond to the two logos that are displayed above the workspace (HPE logo and nautilus). To change these images:

1. Decompress the **.war** file used to install Asset Manager Web Tier in the **AssetManager.war_build** folder.
   It is assumed that you are using the **C:\Program Files\HPE\Asset Manager 9.61 xx\webtier\AssetManager.war** file.
2. Open the **AssetManager.war_build\ext** folder.
3. Create a folder named **images**.
4. Change to this folder.
5. Copy the image(s) to this folder.

   > **Note:** You may need to make sure that your images fit correctly within your Web client's workspace. The dimensions of the images may need to be modified.

6. Rename the file that will appear in the left part of the banner (image that will replace the HPE logo) to **header_left.gif**.
7. Rename the file that will appear in the right part of the banner (image that will replace the nautilus) to **header_right.gif**.
8. Open the **AssetManager.war_build** folder.
9. Select all files and folders.
10. Compress these files and folders in the **AssetManager.war_build\AssetManager.war** file.
11. Copy the **AssetManager.war** file to the **C:\Program Files\HPE\Asset Manager 9.61 xx\webtier** folder (overwrite the previous file).
12. Delete the **webapps\AssetManager.war_build** folder.

13. Reinstall Asset Manager Web Tier. See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section**Installing Asset Manager Web**.

Changing the text of the banner

To change the text of the banner:

1. Decompress the **.war** file used to install Asset Manager Web Tier in the **AssetManager.war_ build** folder.
   It is assumed that you are using the **C:\Program Files\HPE\Asset Manager 9.61 xx\webtier\AssetManager.war** file.
2. Open the **app_labels.properties** file in the **WEB-INF\classes\com\hp\ov\cwc\web** folder with a text editor.
3. Locate the line starting with **App.Name** and specify the text of the banner. For example:

   ```
   App.Name=MY HPE Asset Manager
   ```

4. Locate the line starting with **App.Title** and specify the title of the Web page. For example:

   ```
   App.Title=My HPE Asset Manager Title
   ```

5. Save the **app_labels.properties** file.
6. Reinstall Asset Manager Web Tier. See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Reducing the time it takes to display screens for the first time

Asset Manager Web uses the functions of the Web services to retrieve or modify information in the Asset Manager database.

These functions are described in a WSDL document.

The WSDL document exposes the current database functions as well as the functions that were available at specific moments if the database was tagged at those specific moments.

The functions of the Web services handle objects (fields, screens, actions, and so on) which are called by Asset Manager Web Service via application programming interfaces (API).

These objects are stored in caches (one cache per service).

The first query for a service takes a significant amount of time to execute (around 10 seconds per service) because the cache needs to be loaded during this initial query.

To make pages display faster for Asset Manager Web users, you can set up Asset Manager Web Service to have it preload all or part of the Web services.

However, it will take longer to start your web applications server.

To "preload" one or more services:

1. Edit Asset Manager Web Service's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **WebService.LoadOnStartup** parameter.
3. Modify the parameter's value:

Entry value	`<Version of the Web services to preload>/<Services to preload>, <Version of the Web services to preload>/<Services to preload>, ...` **<Services to preload>**: To include all services, use the asterisk *****. Separate the services using a comma if you want to preload more than one. **<Version of the Web services to preload>** must be the same for all services in the list.
Example	`WebService.LoadOnStartup` In the default value (**Head/***), a request to "preload" all services of the **Head** version is made. For example, if you only want to load the **Administration** version, you must set the parameter value to **Head/Administration**. Separate the services using a comma if you want to preload more than one. For example, to "preload" **Head/Administration** and **Head/Finance**: **Note:** For the list of available services, read the WSDL document see **Tailoring** guide, chapter **Customizing the database**, section **Development best practices/ Tag the Web services/ Displaying the WSDL document**. **Caution:** The Web services version that is specified (**Head** in our example) must be the same as the value of the **WebService.Version** parameter in the

	> Asset Manager Web Tier **package.properties** file (see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**). **Head/Administration, Head/Finance**

4. Reinstall Asset Manager Web Service: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Defining the number of lines to export to Excel

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Exports.Excel.RowCount.Maximum** parameter.
3. Modify the parameter's value:

Entry value	Maximum number of lines to export to Excel
Example	Exports.Excel.RowCount.Maximum=10000

> **Note:** The number of lines to export to Excel can be larger than 65536.

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Defining the number of lines to display in the lists

You can define the number of lines to display in lists for the following types of lists:

- Main list (list of records from a table that is displayed when you click a link on the navigation bar).
- Tab list (list that appears on a tab of a record's details)
- Wizard list (list that appears on a wizard's page)

To define the number of lines to display in the lists:

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Lists.Paging.Count.Maximum.Default** parameter.
3. Modify the parameter's value:

Entry value	Maximum number of lines to display in the lists
Example	Lists.Paging.Count.Maximum.Default=20

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying text read by screen reader software

To modify text that is read by a screen reader in order to identify required fields and links:

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Fields.All.Mandatory.Prompt.Aural** parameter.
3. Modify the parameter's value:

Entry value	Text read by screen reader software to designate a required field or link
Example	Fields.All.Mandatory.Prompt.Aural=Mandatory field or link

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Defining the number of records to display in the drop-down lists

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Fields.LinkEdit.Suggestions.Count** parameter.
3. Modify the parameter's value:

Entry value	Maximum number of lines to display in the lists
Example	Fields.LinkEdit.Suggestions.Count=10

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Choosing the location of the workspace toolbar

The toolbar that is displayed inside the workspace contains buttons used to perform actions on records (**New**, **Modify**, **Delete**, and so on) and the **Actions** drop-down list. The toolbar can be displayed at the top and/or bottom of the workspace.

To choose its location:

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Lists.Toolbars.Display** parameter.
3. Modify the parameter's value:

Entry value	Location of the toolbar: ◦ **header**: The toolbar is located at the top.

	◦ **footer**: The toolbar is located at the bottom. ◦ **both**: The toolbar is located at both the top and bottom.
Example	`Lists.Toolbars.Display=both`

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying how often the cache managing the workflow schemes and instances is refreshed

Information used for a workflow scheme (list and position of boxes and links, and so on) are managed by a **cache** stored in memory on the Asset Manager Web Tier server and refreshed periodically (every **24 hours** by default).

See **Advanced use** guide, chapter **Workflow**, section **Workflow tracking**, paragraph **Refreshing workflow schemas and instances**.

You can modify how often the cache is refreshed. To do this:

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Cache.Workflows.Refresh.Interval** parameter.
3. Modify the parameter's value:

Entry value	Number of seconds between cache refreshes.
Example	**Note: 86400** seconds = 24 hours Cache.Workflows.Refresh.Interval=86400

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Refreshing a detail screen when the cursor leaves a field's or link's edit zone

A detail screen is only refreshed if the field or link that has been modified is referenced by a **Mandatory**, **Read only**, **Irrelevant** or **Default** script of another field or link in the same table.

However, a refresh is not triggered in the following cases:

- Fields whose data type is among the following list:
  - Monetary
  - Text
  - Custom itemized list, if the itemized list is open.
- Scripts that reference a calculated field or a table description string

To force the refresh for exceptions

If you notice that the refresh is not triggered when leaving a field or link for which a refresh should be triggered (unique exception), you can force a refresh:

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Refresh.fields** parameter.
3. Modify the parameter's value:

Entry value	List of fields and links for which a detail screen must be automatically refreshed when the cursor leaves the edit zone. Syntax: `<Truncated SQL name of screen 1>:<Truncated SQL name of field or link 1>|...|<Truncated SQL name of screen n>:<Truncated SQL name of field or link n>` **Truncated SQL name** means the SQL name from which the first two letters, **am**, have been deleted.
Ex	Refresh.fields=Employee:Field1|Employee:Field2

ample	

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the delay before a link's list of values is refreshed

To modify the delay before a link's list of values for auto-completion is refreshed:

Using other parameters to generate the Asset Manager Web Tier log file

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Fields.LinkEdit.AutoCompletion.DelayMS** parameter.
3. Modify the parameter's value:

Entry value	Time in milliseconds before the list of values is refreshed
Example	Fields.LinkEdit.AutoCompletion.DelayMS=700

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **log.properties** parameter.

3. Modify the parameter's value:

Entry value	Full path to the new parameter file **Note:** You must create a new **.properties** file describing how the log file is to be generated, and then reference the new **.properties** file. This file must be referenced using its absolute path.
Example	log.properties=/WEB-INF/classes/log4j.properties

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Using another screen from the **Business home pages** table to display business home pages

To use another screen from the **Business home pages** (amFDView) table to display business home pages:

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **WebService.Object.FunctionalDomainHomePage** parameter.
3. Modify the parameter's value:

Entry value	SQL name of the screen to use
Example	WebService.Object.FunctionalDomainHomePage=FuncDomDisplay

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the style sheet to use to display and print HTML pages from Asset Manager Web Tier

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **alternateTheme** parameter.
3. Modify the parameter's value:

Entry value	Name without the **.css** extension of the new style sheet. **Note:** The style sheet to display the pages must be located in the **ext\css** folder of the **AssetManager.war** file. The style sheet to print the pages must be located in the **ext\css\print** folder of the **AssetManager.war** file.
Example	alternateTheme=am

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying how often the currency cache is refreshed

Asset Manager Web Tier stores the symbol and the position of currencies from the **Currencies** (amCurrency) table in a cache.

This information is used when amounts are displayed.

To modify how often the cache is refreshed:

1. Edit Asset Manager Web Tier's **ehcache.xml** file, which is located in the **C:\Program Files\HPE\Asset Manager 9.61 xx\webtier\AssetManager.war\WEB-INF\classes** folder.
2. Locate the following block:

```
<cache  name="currency"
maxElementsInMemory="1"
eternal="false"
timeToIdleSeconds="21600"
timeToLiveSeconds="21600"
overflowToDisk="false"
diskPersistent="false"
memoryStoreEvictionPolicy="LRU"/>
```

3. Modify the value of the **timeToLiveSeconds** parameter to set the cache refresh frequency of currency.
4. Save the **ehcache.xml** file.

Modifying how often the itemized lists cache is refreshed

Asset Manager Web Tier stores itemized list values from the **Itemized lists** (amItemizedList) table in a cache.

This information is used when completing a field that is populated via a non-system itemized list.

To modify how often the cache is refreshed:

1. Edit Asset Manager Web Tier's **ehcache.xml** file, which is located in the **C:\Program Files\HPE\Asset Manager 9.61 xx\webtier\AssetManager.war\WEB-INF\classes** folder.
2. Locate the following block:

```
<cache  name="openItemizedList"
maxElementsInMemory="100"
eternal="false"
timeToIdleSeconds="1200"
timeToLiveSeconds="1200"
overflowToDisk="false"
diskPersistent="false"
memoryStoreEvictionPolicy="LRU"/>
```

3. Modify the value of the **timeToLiveSeconds** parameter to set the cache refresh frequency of itemized lists.
4. Save the **ehcache.xml** file.

Enabling the confirmation page before a **script** type action is executed

Asset Manager Web Tier stores itemized list values from the **Itemized lists** (amItemizedList) table in a cache.

This information is used when completing a field that is populated via a non-system itemized list.

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Actions.Scripts.Launch.Confirmation** parameter.
3. Modify the parameter's value:

Entry value	◦ **true**: Enables confirmation message ◦ **false**: Disables confirmation message
Example	Actions.Scripts.Launch.Confirmation=true

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the maximum number of segments to display in report or statistics graphs

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Charts.Categories.Count.Maximum** parameter.
3. Modify the parameter's value:

Entry value	Maximum number of segments to display
Example	Charts.Categories.Count.Maximum=10

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the maximum number of reports to display in dashboards

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **Dashboards.Items.Count.Maximum** parameter.
3. Modify the parameter's value:

Entry value	Maximum number of reports
Example	Dashboards.Items.Count.Maximum=6

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the maximum number of segments to display in graphs for business home pages

To modify the maximum number of segments to display in report or statistic graphs when they are displayed in a business home page:

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **BusinessHomePages.Statistics.Count.Maximum** parameter.

3. Modify the parameter's value:

Entry value	Maximum number of reports
Example	BusinessHomePages.Statistics.Count.Maximum=10

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the maximum number of statistics to display in business home pages

1. Edit Asset Manager Web Tier's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **BusinessHomePages.Items.Count.Maximum** parameter.
3. Modify the parameter's value:

Entry value	Maximum number of reports
Example	BusinessHomePages.Items.Count.Maximum=6

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the root used for URLs in Asset Manager Web Service

To modify the **<Root>** used for Asset Manager Web Service URLs (**<http://<Server name>:<Port>/<Root>**):

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section

Installing Asset Manager Web.

2. Locate the **context.root** parameter.
3. Modify the parameter's value:

Entry value	New name to use
Example	context.root=HPAssetManager

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the root used for URLs in Asset Manager Web Service

> **Caution:** This is only possible if WebSphere Application Server or Oracle WebLogic is used as your application server (not Tomcat).

To modify the **<Root>** used for Asset Manager Web Service URLs (**<http://<Server name>:<Port>/<Root>**):

1. Edit Asset Manager Web Service's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **context.root** parameter.
3. Modify the parameter's value:

Entry value	New name to use
Example	context.root=HPAssetManager

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the root used for URLs in Asset Manager Web Tier

> **Caution:** This is only possible if WebSphere Application Server or Oracle WebLogic is used as your application server (not Tomcat).

To modify the **<Root>** used for Asset Manager Web Tier URLs (**<http://<Server name>:<Port>/<Root>**):

1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **context.root** parameter.
3. Modify the parameter's value:

Entry value	New name to use
Example	`context.root=HPAssetManager`

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the name of Asset Manager Web Service that is displayed in administration screens of your application server

> **Caution:** This is only possible if WebSphere Application Server or Oracle WebLogic is used as your application server (not Tomcat).

1. Edit Asset Manager Web Service's **package.properties** file: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section

Installing Asset Manager Web.

2. Locate the **display.name** parameter.
3. Modify the parameter's value:

Entry value	New name to use
Example	`display.name=HPAssetManager`

4. Reinstall Asset Manager Web Tier: see **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Modifying the name of Asset Manager Web Tier that is displayed in administration screens of your application server

> **Caution:** This is only possible if WebSphere Application Server or Oracle WebLogic is used as your application server (not Tomcat).

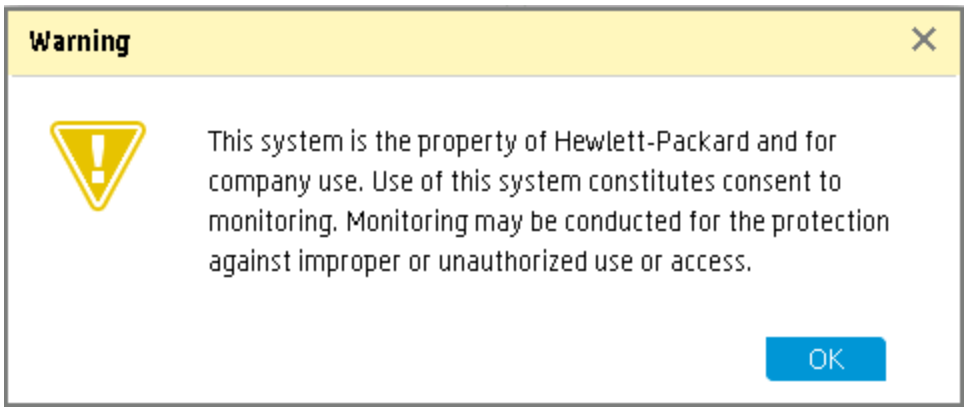
1. Edit Asset Manager Web Tier's **package.properties** file: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.
2. Locate the **display.name** parameter.
3. Modify the parameter's value:

Entry value	New name to use
Example	`display.name=HPAssetManager`

4. Reinstall Asset Manager Web Tier: See **Installation and upgrade** guide, chapter **Installing, configuring, removing and updating Asset Manager Web**, section **Installing Asset Manager Web**.

Creating a logon disclaimer for the Web client

You can create a disclaimer that appears when a user logs on to the Web client. The content of the disclaimer can be customized.

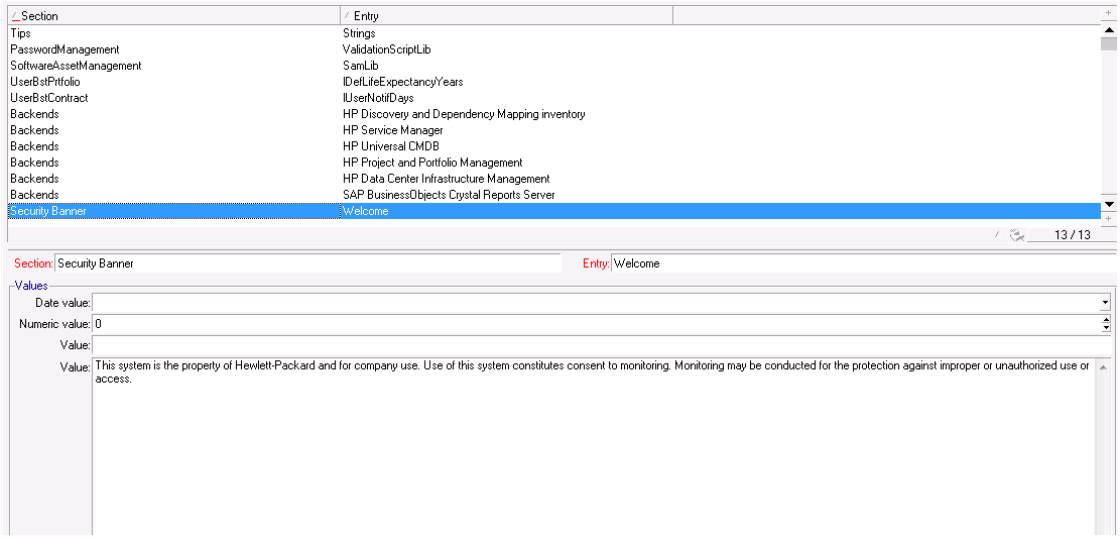


To do this, follow these steps:

1. Open the Windows client.
2. On the **Administration** menu, click **List of screens**.
3. Open **Options of the application (amOption)**.
4. Click **New** to create a new record.
5. Enter *Security Banner* in the **Section** box, then enter *Welcome* in the **Entry** box.

   > **Note:** To use this feature, you must not change the value of the **Section** and the **Entry** box.

6. In the second **Value** box, enter the text you want to be displayed in the logon disclaimer.



7. Click **Create**.
8. Log on to the Web client and verify that the logon disclaimer appears.

User defined customizations

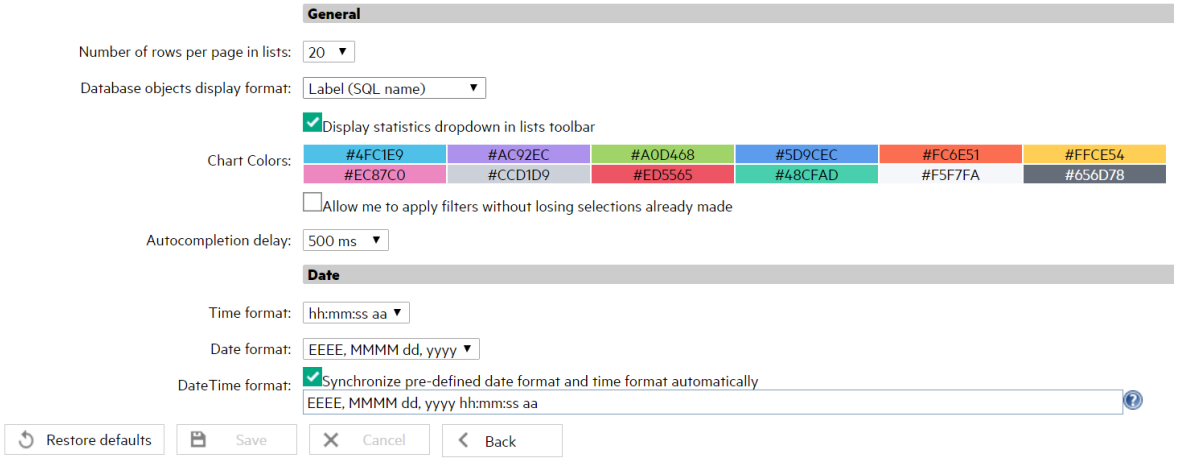
This section includes:

Preferences pane 155
Customizing Chart Colors 158
Configuring list columns 158
Editing the package.properties file 159

Preferences pane

You can customize a number of display features for the current user, using the **My Preferences** pane. To display this, click your user name at the upper-right corner of the Asset Manager window, and then click **My Preference**:

The **My Preferences** pane is shown:



To customize any of the following features:

Feature	Details
General	
Display statistics dropdown in lists toolbar	Displayed by default. Uncheck to hide this dropdown.
Number of rows per page in list	Defaults to 20. (This global default can itself be changed by the Administrator.)
Database objects display format	The default is **Label (SQL name)**. The options are: • Label (SQL name) • SQL name • SQL name (Description) • SQL name (Label) • Label • Description • Description (SQL name)
Allow me to apply filters without losing selections already made	Selecting the box enables the user to choose whether to keep or clear the selections already made before applying a filter.
Chart Colors	Explained in detail in the following section.
Autocompletion delay	Defaults to 500 ms. Available options: 500 ms, 1500 ms and 3000 ms.
Date	

Feature	Details
Time format	Select the time display format. The default value is hh:mm:ss aa. The options are: • h:mm:ss aa: display the time in 12-hour format with the initial digit of the hour omitted. For example: 9:30:45 am • hh:mm:ss aa: display the time in 12-hour format with the initial digit of the hour reserved. For example: 09:30:45 am • H:mm:ss: display the time in 24-hour format with the initial digit of the hour omitted. For example: 9:30:45 • HH:mm:ss: display the time in 24-hour format with the initial digit of the hour reserved. For example: 09:30:45
Date format	Select the date display format. The default value is EEEE, MMMM dd, yyyy. The options are: • M/d/yy • M/d/yyyy • MM/dd/yy • MM/dd/yyyy • yy/MM/dd • yyyy-MM-dd • dd-MMM-yy • EEEE, MMMM dd, yyyy • MMMM dd, yyyy • EEEE, dd MMMM, yyyy • dd MMMM, yyyy
DateTime format	• If you select the **Synchronize pre-defined date format and time format automatically** option, the DateTime format field is read-only and automatically populated with the combination of your selection for **Time format** and **Date format**. • If you deselect this option, you can enter a customized value for DateTime format. **Tip:** You can click ? for the recognized letters, separators and patterns in the date/time format definition.

To return all these features to their system default values, click **Restore defaults**.

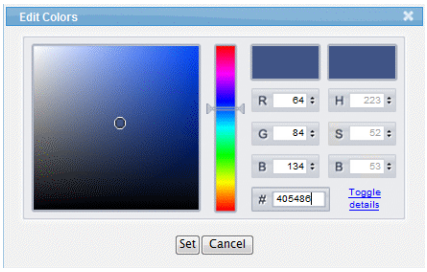
To save your customizations or restored defaults, click **Save**.

Note: The default values of these user preferences are defined in the **default_user_**

> **pref.properties** file located in **<Tomcat installation folder>\webapps\AssetManager\WEB-INF\classes\com\hp\ov\ac\web\bean**.

Customizing Chart Colors

The **Preferences** pane provides access to a palette of twelve default **Chart Colors**, for which the hex values are shown against the background color. To change any one of these colors, enter edit mode by clicking the **Modify** button, then double-click in the corresponding color box to display the **Edit Colors** tool, as shown below.



There are three ways to select a new color:

- Use the central slider bar to choose an approximate color in the square selector box on the left of the pane. Then click or drag the dropper to select a color. The two smaller boxes on the upper right of the pane show a sample (left) of the selection and the original color (right) that is displayed before the **Edit Colors** tool opens.
- Enter RGB values between 1 and 255.
- Enter a hex value in the # field.

Click **Set** to apply your selected color to the chart palette on the **Preferences** pane.

Click **Save** to save your customized chart color(s) and any other preferences that you have changed.

> **Note:** You can also customize chart colors using a script. This method applies to both Web and Windows clients. For more details, see section **Customizing Windows clients**, chapter **Customizing a Windows client**, section **User defined colors in scripted queries for charted statistics**.

Configuring list columns

To customize which columns display for the current list, and in which order:

1. Click **More** > **Configure the columns of the list**.

   Or, click the icon on the tabs of the details pane.

2. The columns customization pane displays. The following example is the the list of columns for the **Portfolio Items** (amPortfolio) list:

   Use the arrow buttons in the middle to transfer columns to and from the the lists of **Available Columns** and **Selected Columns** (displayed).
   Use the arrow buttons to the right of **Selected Columns** to change the order of the columns as they appear in the list.

3. To return all these features to their system default values, click **Restore defaults**.
4. To save your customizations or restored defaults, click **Save**.

Editing the **package.properties** file

Parameters used for configuring several aspects of the Asset Manager web client are found in the Asset Manager Web **web.xml** file which is stored in the zipped application **war** file.

For deployment purposes the values of these parameters are actually specified and can be edited in the **package.properties** file for your web server. These are explained below.

> **Note:** In version 5.20 of Asset Manager Web the format of the **web.xml** file has changed, and the parameters it holds have undergone a major overhaul.
>
> - Old parameters, used to configure earlier versions of of Asset Manager Web will no longer be recognized.
> - Where an old parameter has a new equivalent, values have often switched polarity or meaning. For these reasons, a search and replace on parameter names is not an adequate way of migrating your configuration data. Instead, you should re-examine all the settings stored in the **package.parameters** file.

> **Caution:** You should never edit the **web.xml** file directly. Rather, edit the parameters in the **package.properties** file and use the deployment script (**deploy.bat** or **deploy.sh**) to deploy them. For more details see the **Installation and Upgrade** guide, chapter **Installing, configuring, removing updating Asset Manager Web**, section **Installing Asset Manager Web**.

Commonly customized parameters

These parameters are those which you are more likely to customize:

Parameter name	Explanation
WebService.EndPoint.SOAP	URL of the SOAP Web service.
WebService.EndPoint.REST	URL of the REST Web service.
WebService.Version	Web service version name.
Exports.Excel.RowCount.Maximum	Max number of rows to export to Excel.
Fields.All.Mandatory.Prompt.Aural	Aural prompt for mandatory fields that are visually indicated by an asterisk. The English default, used when this property is left blank, is "Mandatory field". A default with a similar meaning is provided in other languages. Because a default is used in the event that this property is left blank, the prompt cannot be removed. This is intentional. In any event this prompt will not be seen or heard by users who are not using a screen reader.
Fields.LinkEdit.AutoCompletion.DelayMS	Delay in milliseconds after keypress before autocompletion in LinkEdit controls.
Fields.LinkEdit.Suggestions.Count	LinkEdit control popup size.
Fields.TableChoiceList.AutoCompletion.DelayMS	Delay in milliseconds after keypress before autocompletion in TableChoiceList controls.
Fields.TableChoiceList.Suggestions.Count.Maximum	TableChoiceList control popup maximum size. A negative or zero value provides for an unlimited scrolling list.
Lists.Paging.Count.Maximum.Default	Default value of user preference for the maximum number of documents to show per page in a list.
Lists.Toolbars.Display	Sets how the list toolbar is to be displayed relative to lists. Recognized values are "header" (above list), "footer" (below list) and "both" (above and below list).

Less commonly customized parameters

The following parameters can also be customized:

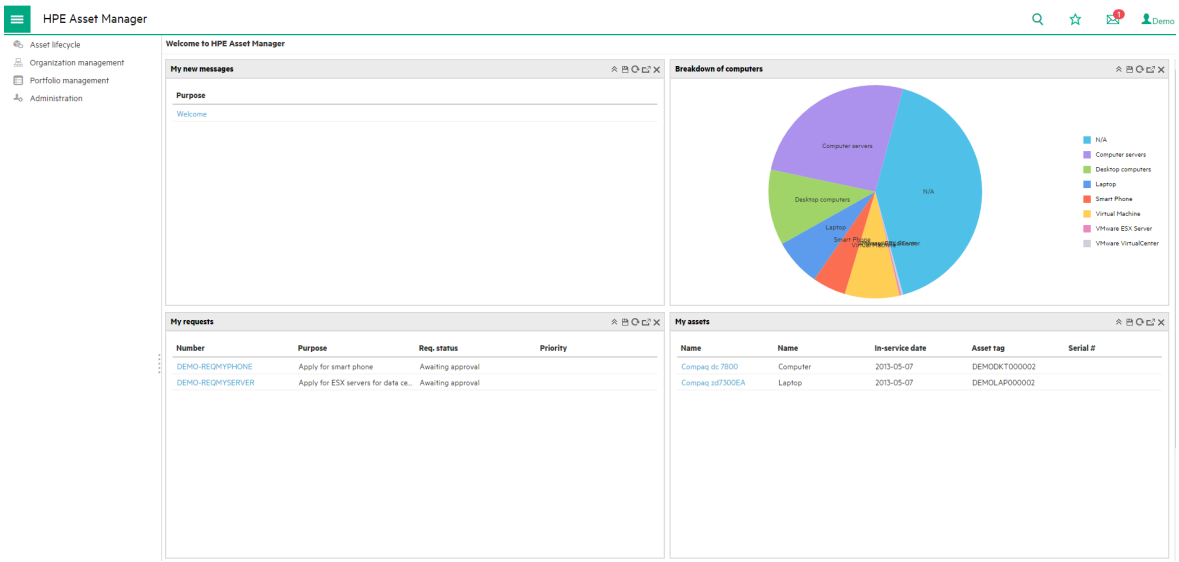
Parameter name	Explanation
Actions.Scripts.Launch.Confirmation	Whether or not to show a confirmation screen before launching a script action.
BusinessHomePages.Items.Count.Maximum	Maximum number of items to display on a Business Home Page.
BusinessHomePages.Statistics.Count.Maximum	Maximum number of statistical breakdowns to display on a Business Home Page.
Cache.Workflows.Refresh.Interval	Workflow refresh interval in seconds (86400 seconds = 24 hours).
Charts.Categories.Count.Maximum	Max number of categories (e.g. bars) to display on a chart.
Charts.General.Height	General statistics chart height (pixels).
Charts.General.Width	General statistics chart width (pixels).
Charts.Dashboards.Height	Statistics chart height in Dashboards (pixels).
Charts.Dashboards.Width	Statistics chart width in Dashboards (pixels).
Charts.StatisticsScreens.Height	Statistics chart height in Statistics screens (pixels).
Charts.StatisticsScreens.Width	Statistics chart width in Statistics screens (pixels).
Charts.Wizards.Height	Statistics chart height in Wizards (pixels).
Charts.Wizards.Width	Statistics chart width in Wizards (pixels).
Dashboards.Items.Count.Maximum	Max number of items per dashboard.
Documents.Downloads.OpenInBrowser.AllowAll	Whether or not to globally permit documents to be opened within the browser window.
Documents.Downloads.OpenInBrowser.AllowedFileTypes	Which file types to allow being opened within the browser window. This value is

Parameter name	Explanation
	only used if Documents.OpenInBrowser.AllowAll=false
Fields.ColorChooser.Font.White.ColorDarknessLevel	For color choosers, the color value is displayed in white for darker colors. This parameter determines the minimum darkness (0-1) at which the color value is displayed in white rather than black (0 is darkest, 1 is lightest).
Workflows.Diagrams.Captions.Font	For canvas caption font.
WebService.Service.Authentication	Web service used for authentication.
WebService.Service.Currency	Currency service provider - Uses the amCurrency screen in the amCurrency table.
WebService.Service.Dashboard	Dashboard service provider - Uses the amDashboardItem screen in the amDashboardItem table.
WebService.Object.Dashboard	Functional Domain Homepage object - Uses the amDashboardItem screen in the amDashboardItem table.
WebService.Service.DateAlarm	DateAlarm service provider - Uses the amDateAlarm screen in the amDateAlarm table.
WebService.Service.DocBlob	File upload/download service provider - Uses the amDocBlob screen in the amDocBlob table.
WebService.Service.FunctionalDomainHomePage	Functional domain service provider - Uses the FuncDomainHomePage screen in the amFuncDomain table.
WebService.Object.FunctionalDomainHomePage	Functional Domain Homepage object - Uses the FuncDomainHomePage screen in the amFuncDomain table.
WebService.Service.FunctionalDomain	Functional domain service provider - Uses the amFuncDomain screen in the amFuncDomain table.
WebService.Service.ItemizedList	temizedList service provider - Uses the amItemListVal screen in the

Parameter name	Explanation
	amItemListVal table.
WebService.Service.Mail	Mail service provider - Uses the amMail screen in the amMail table.
WebService.Service.Profile	Profile service provider - Uses the amMasterProfile screen in the amMasterProfile table.
WebService.Service.ReadMailWizard	Read mail wizard service provider - Uses the sysCoreReadMsg wizard.
WebService.Service.Reports	Reports service provider - Uses the amReports screen in the amReports table.
WebService.Service.WorkflowSchema	Workflow schema service provider.
AssetManager.WS.EmplDeptService (no longer used)	Employee service provider.

Homepage of the Web client

The homepage of the Web client consists of Asset Manager statistical widgets. The data, type, and layout of the widgets are all customizable.



A widget that has drill-down capabilities can drill down to either a new widget (linked widget) or the detailed items in the widget. Asset Manager enables end users to drill down from a widget into another one at several levels. Drill-down can be defined on respective columns or clickable areas of the chart.

This section includes:

How to create a homepage widget 164
Personalizing the homepage 184
User profiles 184
How to create a homepage template 186
Importing/exporting homepage widgets and templates 186

How to create a homepage widget

Before you add a widget to the homepage, you need to create and configure the data source of the widget. To do this, follow these steps.

> **Note:** A user with the **Homepage widgets management** profile can create homepage widgets. For more details of the **Homepage widgets management** profile, see "User profiles" on page 184.

Creating an AQL widget

This section describes how to create an AQL widget.

Adding an AQL widget

1. On the navigation bar, expand **Administration**, and then click **Home page widgets**.
2. In the left-hand side pane, the + button, and then click **AQL Report**.
3. In the **Name** box, type the widget name.
4. Select one of the following statistic types:
   - : Data list
   - : Pie chart

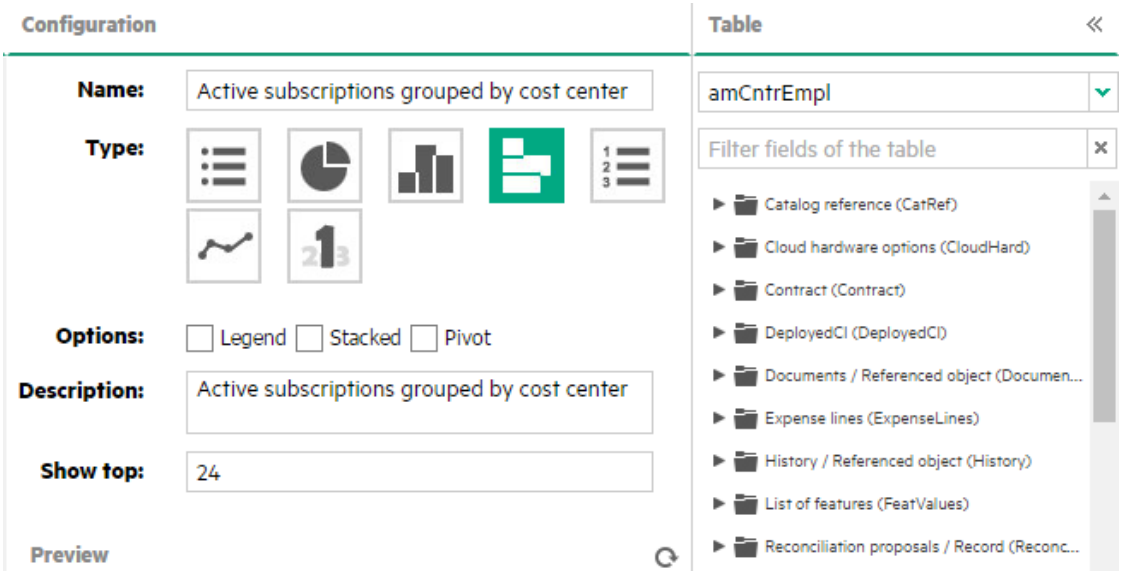
- : Column chart
- : Bar chart
- : Number chart
- : Line chart
- : Big solo number chart

5. Select one or more of the statistic options according to your requirement:
   - Legend
   - Stacked
   - Pivot
   - Number of lines to display (Show top)
6. From the drop-down list in the middle of the page, select or type the Asset Manager table of the data source.

Configuring the AQL widget

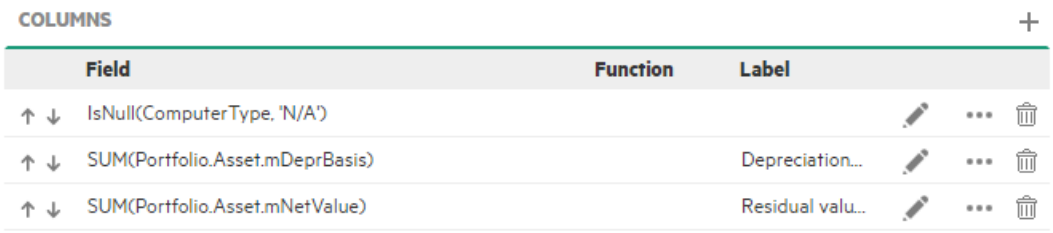
1. After you select a table for the widget, all fields and links of this table are displayed. You can use any of these fields or links to create an AQL statement. The result of the AQL statement is displayed in the form of the statistic type you select.

   > **Note:** You can use the **Filter fields of the table** box to search for a field or a link.



2. Specify the selected column of the AQL statement.
   a. Click a field or link in the fields/links pane, and then click **Select as COLUMN**. The field or link will be added to the **COLUMNS** box.
   b. You can also add a custom column to the **COLUMNS** box by clicking the "+" button. A custom column does not have to be a field or the link in the table, a custom column can be anything that complies to the AQL syntax. For example, a custom column can be an expression.

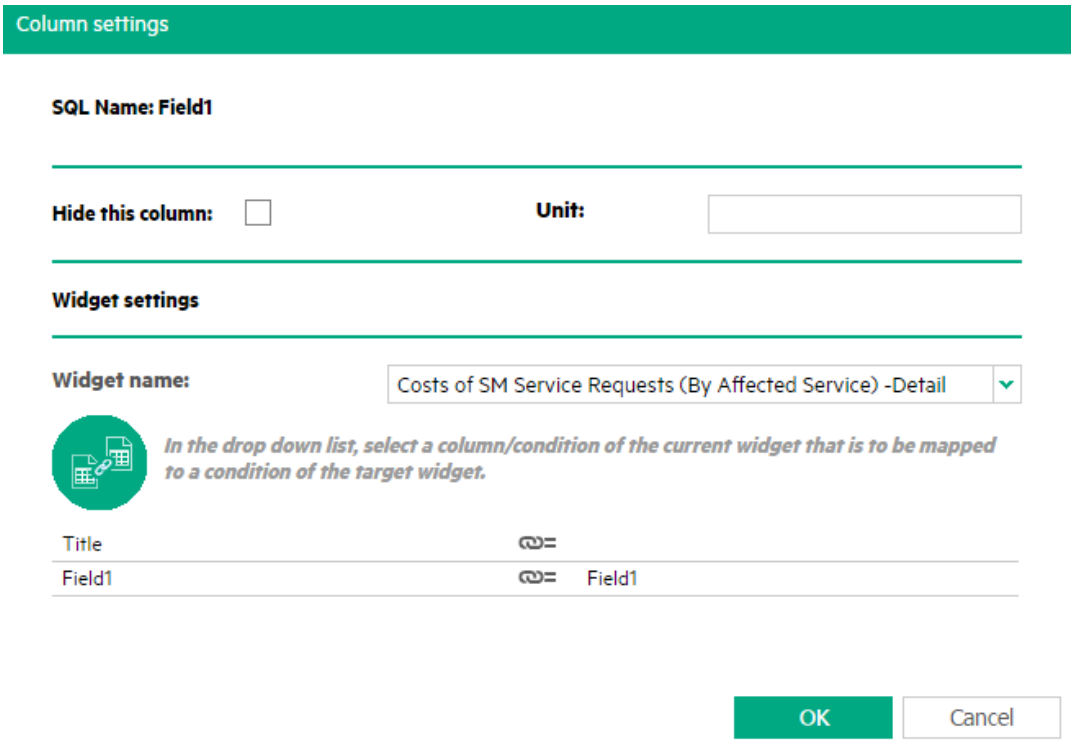
      Example of a field that is an expression:



      If the column is a field or expression (not a link to another report), the edit control looks like .

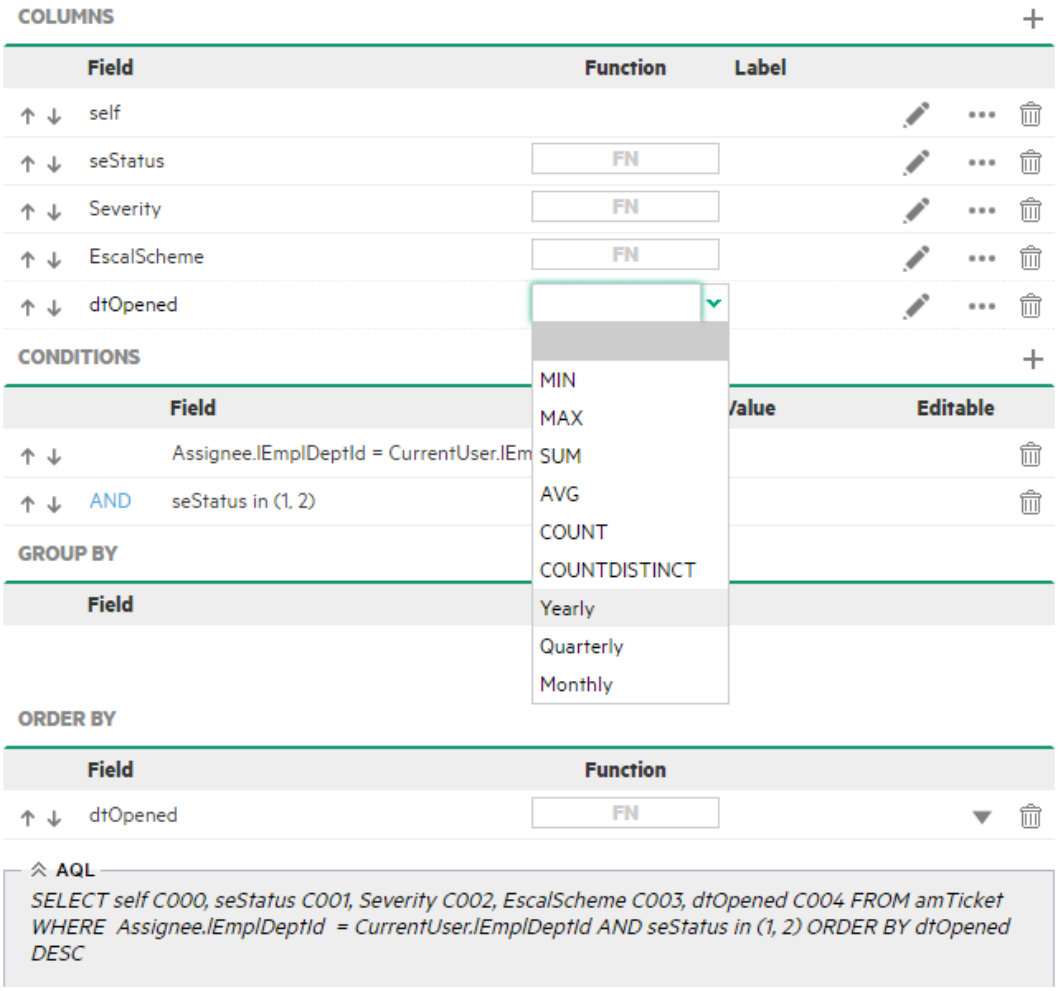
      If the column is a link to another linked report, the edit control looks like .

      If you do not want this column to be displayed in the widget, check the **Hide this column** checkbox.



> **Note:** If you want to use a link in a custom column, do not use the link directly. Instead, you must append the field name to the link. For example, do not use *Company + 'Ltd'*. Instead, use *Company.name + 'Ltd'*.

c. You can select a function for the column in the **Function** drop-down list. If the column type is **Datetime**, the following three functions are also available: "Monthly", "Quarterly", "Yearly".



> **Note:** In a custom column, the **Function** drop-down list is not available, you must manually enter the function with the correct syntax. For example, you cannot select the SUM function from the **Function** drop-down list. Instead, you must manually enter the function like "SUM(*field*)".

d. You can enter a label for the column in the **Label** field. If you enter a label, the selected column is displayed with the label in the chart. If you do not enter a label, the display name of the column is used in the chart.

e. The Asset Manager widget allows drill-down. To modify the behavior when clicking the corresponding area in the widget, click the button.

> **Note:** If you do not specify a linked report for the column, you will be redirected to the detail list when you click the corresponding part in the widget.

i. The **Column settings** window pops up.

ii. In the **Linked report settings** drop-down list, select a widget you want to link to.

iii. After you select a widget, the conditions of the target widget appear at the bottom of the **Column settings** window.

iv. In the **Select a Column or Condition** drop-down list, select a column or condition of the current widget that is to be linked to the condition of the target widget.

v. After a link is created between two widgets, when you click the selected column in the current widget, you will be redirected to the target widget. The value you selected in step iv is used as the value of the condition of the target widget.

f. You can click the ••• icon to see several possible options :

- Add a selected column to the conditions.

  > **Note:** We recommend that you do not add a link to the conditions directly. Instead, specify the field of the link. For example, use *Company.name* instead of *Company*.

- Have the chart ordered by the selected column.

  > **Note:** A custom column cannot be used in the **ORDER BY** clause.

- Have the chart grouped by the selected column.

  > **Note:** A custom column or a link cannot be used in the **GROUP BY** clause.

g. Repeat step a to step e to specify more selected columns.

h. You can also create a custom column by click the + button in the upper-right corner of the pane. Notice that a custom column cannot be linked to another widget, it does not have a button.

i. After you create all the selected columns, you can click the ↑ button or the ↓ button to change the order of the selected columns.

3. Specify the condition of the AQL statement.

a. Click to a field or link in the fields/links pane, and then click **Select as CONDITION**. The field or link will be added to the **CONDITIONS** box.

b. You can also create a custom condition by click the + button in the upper-right corner of the pane.

> **Note:** When creating a custom condition, you do not need to add quotes for the strings.

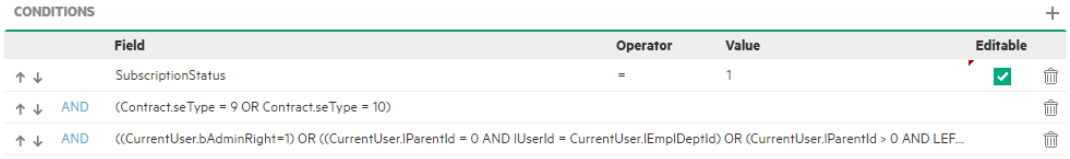
> The quotes are added automatically in the AQL statement.

c. Select the operator and the value of this condition.

d. If you want the condition to be editable on the homepage by click the button of the widget, check the **Editable** check box.

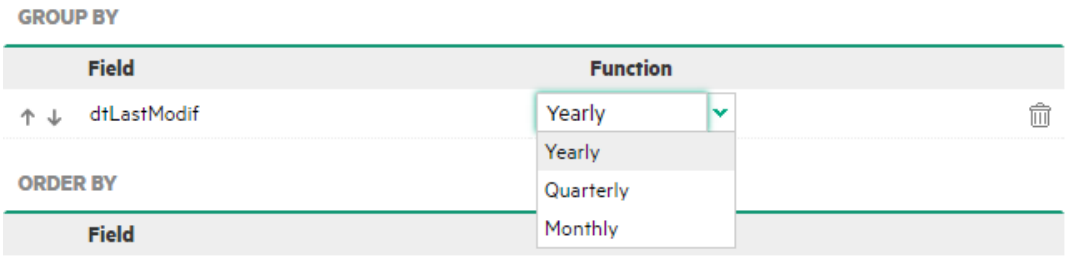
e. Repeat step a to step c to create new conditions. When you add new conditions, you must specify AND/OR in the first field.

f. After you create all the conditions, you can click the button or the button to change the order of the conditions.



4. Specify the "Group by" clause of the AQL statement.
   a. If you want the result to be grouped by a field, click the field in the fields pane, and then click **Select as GROUP BY**. The field will be added to the **GROUP BY** box.

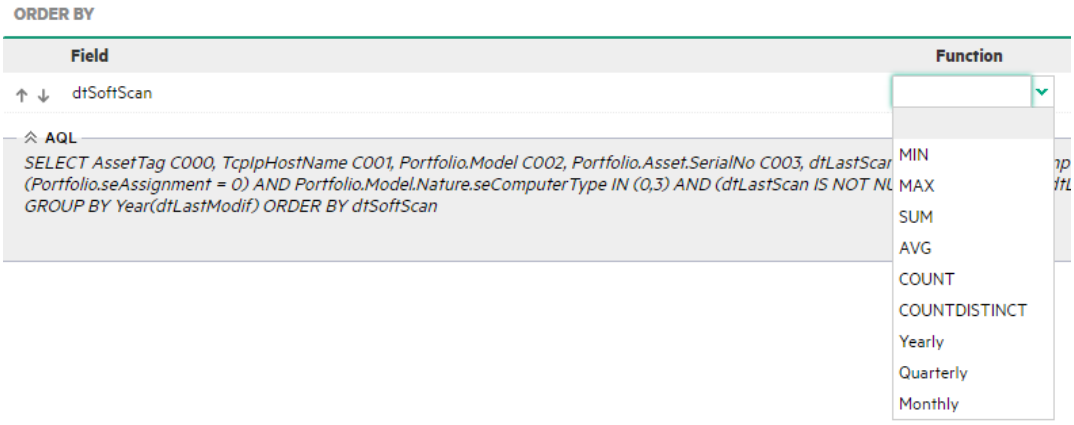
   > **Note:** When add a field of the **Date** type or **Datetime** type to the "Group by" clause, you can select the granularity of date (Monthly/Quarterly/Yearly) in the **Function** drop-down list next to the field.



   b. After you create all the GROUP BY fields, you can click the button or the button to change the order of the GROUP BY fields.

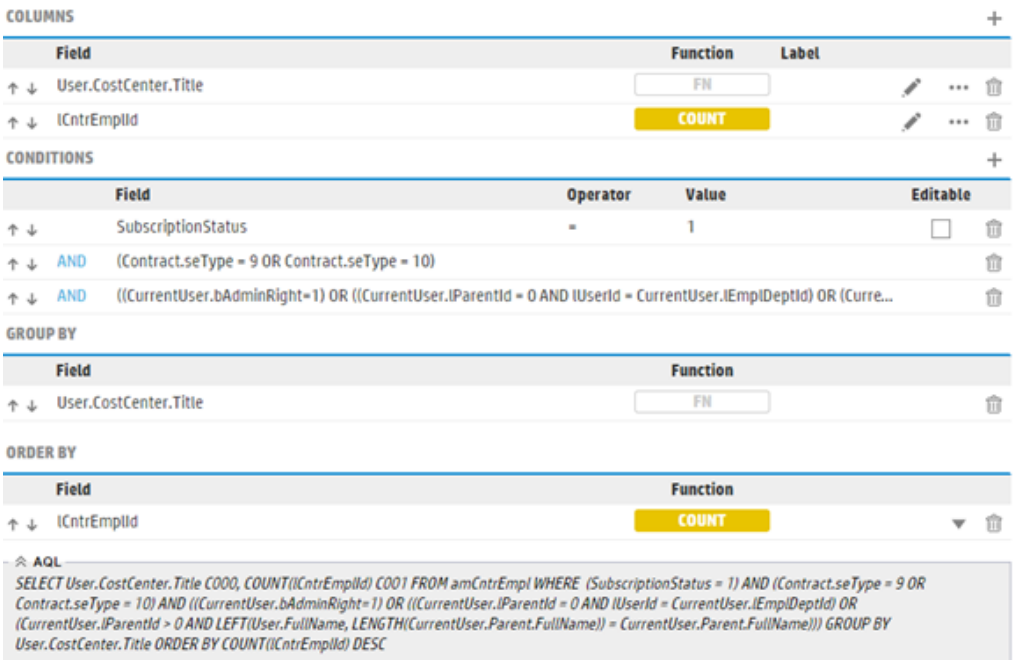
5. Specify the "Order by" clause of the AQL statement.
   a. If you want the result to be ordered by a field or link, click the field or link in the fields/links pane, and then click **Select as ORDER BY**. The field or link will be added to the **ORDER BY** box. Alternatively, if the field or link is used as a selected column, you can do this by checking the **Order** check box in the **SELECT** box.

   b. You can specify a function in the order by clause next to each field.



After you create all the ORDER BY fields, you can click the ↑ button or the ↓ button to change the order of the ORDER BY fields.

Note: When you add a field to the GROUP BY clause, the fields that can be specified in ORDER BY must be contained either in the GROUP BY clause or in an aggregate function (usually in the selected columns).



Generating the AQL widget

1. Verify the AQL statement that is generated automatically in the bottom-right corner. Point to the **AQL** box, you will see an error message if the AQL statement has any errors. Modify the settings in step 2 to correct the AQL statement.

   > **Note:** You cannot directly modify the AQL statement in the **AQL** box, it can only be generated by the settings in step 2.

2. Click **Preview**, make sure the chart contains the information you wanted.
3. Click **Save**.

Unit

The unit of a column is specified by one of the following methods.

- You can click the icon of the selected column to specify the unit. For example, you can type *KG* in the **Unit** box, the unit is appended to the label or the data itself in the chart.
- If you do not manually specify the unit. Asset Manager determines the unit automatically.
  - For the **Datetime** data type, the display format is determined by the system locale settings.
  - For a monetary column, the unit is determined by its **Ref1** and **Ref2** fields.
  - For a number column, the display format is determined by the system locale settings. For example, if the locale is set to the United States, 100000 is displayed as 100,000.

Creating a script widget

Besides AQL widgets, you can create complex widgets with Basic scripts. This section describes how to create a script widget.

Adding a script widget

1. On the navigation bar, expand **Administration**, and then click **Home page widgets**.
2. In the left-hand side pane, the + button, and then click **Script Report**.

3. In the **Name** box, type the widget name.
4. Select one of the following statistic types:
   - : Data list
   - : Pie chart
   - : Column chart
   - : Bar chart
   - : Number chart
   - : Line chart
   - : Gauge chart
   - : Big solo number chart
   -

: SAM chart
5. Select one or more of the statistic options according to your requirement:
   - Legend
   - Stacked
   - Pivot

Configuring the script

This sections describes how Asset Manager parses the script and creates the widget.

Data source of the script widget

The data source of the script widget is a JSON object, which contains two array objects, **Schema** and **Result**.

The **Schema** object defines the items to be displayed in the widget with the following properties.

- **displayLabel**

  If you set a value for this property, the value is shown as a display label in the widget. If you do not set this value, the SQL name is shown as the display label. If you do not set a value for this property and the SQL name does not exist, the column name such as "C000" is shown as the display label.

- **unit**

  The unit of the current item.

- **sqlName**

  If you want to enable drill-down from the current report to a list in the navigation tree, you can set this property as the SQL name of the field to be linked.

  > **Note:** In addition, to enable drill-down to a list, you also need to set a parameter named tableName, the value of tableName is a table name in the AM database. For more information, see "Drill-down in the script widget" on page 176.

- **type**

  The type (String, Long, Double, Date) of the item. if you want to enable drill-down, you must set this property.

- **userType**

  How the item is displayed (System Itemized List, Money, Percentage). Leave this property empty if none of these three fits.

The **Result** object consists of the values of the items to be displayed in the widget.

The following example shows a typical JSON object that can be used as the data source of a script object.

```
{
  "Schema":[
  {
    "C000": {
       "displayLabel":"",
       "unit": "",              // not mandatory.
       "sqlName": "",           // not mandatory.
```

```
      "type": "",
      "userType": ""          // not mandatory.
    }
  },
  {
    "C001":{
      "displayLabel":"",
      "unit": "",             // not mandatory.
      "sqlName": "",          // not mandatory.
      "type": "",
      "userType": ""          // not mandatory.
      }
  },
  ...
  ],
  "Result":[
  {
    "C000": "",
    "C001": ""
  },
  ...
  ]
}
```

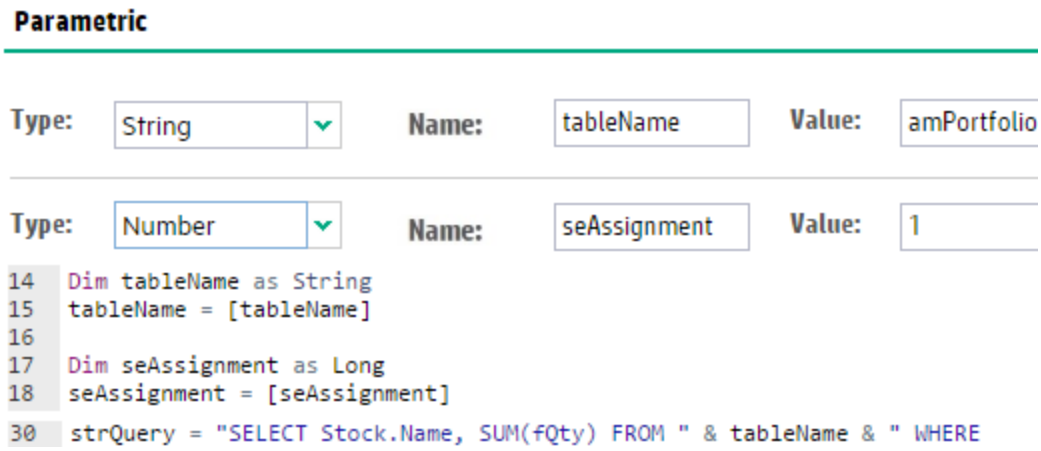
Parameters of the script widget

Besides the properties of the JSON object, you can also create parameters for the script widget.

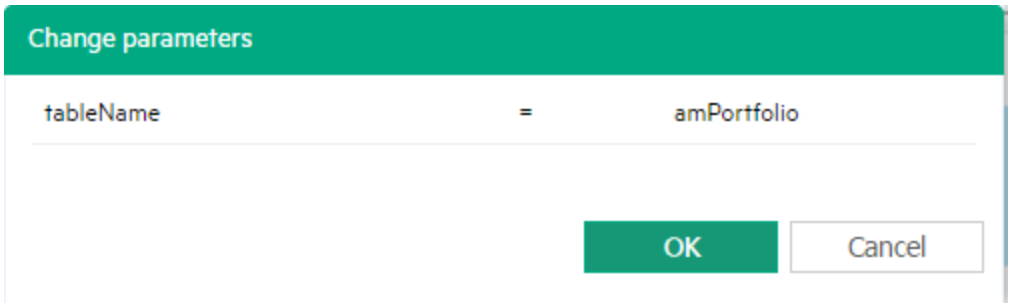
To create a parameter, follow these steps.

1. Click ⚙ to open the **Configuration** window.
2. Click + to add a new parameter.
3. Click **OK**.
4. Use the parameter in the script.

For example, parameters can be used as shown below.



If the widget has been added to the homepage, you can modify the parameter on the homepage by clicking ⚙.



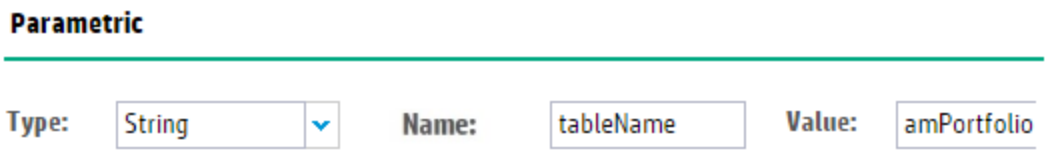
Drill-down in the script widget

This section describes how to enable drill-down in the script widget.

Drill-down to a list on the navigation bar

To enable drill-down to a list on the navigation bar, you must create a parameter named **tableName** for the widget. To do this, follow these steps.

1. Click ⚙ to open the **Configuration** window.
2. Click + to add a new parameter.
3. Create the **tableName** parameter as follows.



4. Click **OK**.
5. Configure the **sqlName** and **unit** properties in the JSON object.

```
"Schema": [{
    "C000": {
        "unit": "",
        "displayLabel": "Stock",
        "sqlName": "Stock.Name",
        "type": "String",
        "userType": ""
    }
```

Drill-down to an existing widget

To enable drill-down to an existing widget, follow these steps.

1. Click ⚙ to open the **Configuration** window.
2. In the **Linked report settings** drop-down list, select a widget you want to link to.
3. After you select a widget, the conditions of the target widget appear at the bottom of the **Column settings** window.
4. In the **Select a Column or Condition** box, enter the name of a column of the script. For example, enter C000, C001, C002, and so on. The value of the column will be used as the condition of the widget you link to.
5. Click **OK**.

Writing a script to generate the JSON object

You can write a VBScript to generate the JSON object to be displayed in the widget. To ease the developing procedure, Asset Manager provides you with the following widget APIs.

Function: createSAMWidgetSTDSchema() As String

Description:

This function is used to create the standard schema of the widget, it can be used only when the widget type is SAM. It contains 4 nodes:

- C000 - Stores the label of the SAM widget.
- C001 - Stores the major value of the SAM widget. It must be a long type value and will be shown as a big and bold number.
- C002 - Stores the description label besides the major value.
- C003 - Stores the color of the big number.

Function: initScriptWidgetSchema() As String

Description:

This function is used to create the standard schema for a normal widget (non-SAM type widget). It returns the standard JSON schema. The format is shown as follows:

```
{
"Result": [],
"Schema": []
}
```

Function: addScriptWidgetSchemaNode(schema As String, nodeId As String, label As String, valType As String, valUnit As String, userType As String, sqlName As String) As String

This function is used to add custom nodes to all kinds of JSON schema.

Parameters:

- schema - The JSON metadata.
- nodeId - The node ID. For example, "C004", "C005", and so on.
- label - The display label for the node.
- valType - The data type of the node. For example, Long, String, and so on.
- valUnit – The unit of the node's value.
- userType – For specific data types. The values can be System Itemized List, Money, Percentage.
- sqlName – The field name in the table. DrillDown function will use the field name and value as conditions in URL to pass to the detail screen. It should be used together with the configuration parameter 'tableName'.

> **Note:** Before invoking this function, you must call **createSAMWidgetSTDSchema()** or **initScriptWidgetSchema()** to generate the JSON schema.

Function: assembleNodeAttr(nodeSchema As String, nodeId As String, attrVal As Variant) As String

Description: This function is used to add attribute value to an existing attribute key-value list. For example: "C004":123,"C005":"abc".

Parameters:

- nodeSchema - The existing attribute key-value list.
- nodeId - The node ID. For example: "C004", "C005", and so on.
- attrVal - The attribute value. For example: 123, abc, and so on.

Function: addScriptWidgetNodeValue(schema As String, nodeVal As String) As String

Description: This function is used to add node value to widget JSON schema.

Parameters:

- schema - The JSON metadata.
- nodeVal - A set of key-value mappings.

Function: addScriptWidgetSchemaNodeEx(schema As String, nodeId As String, label As String, valType As String, valUnit As String, userType As String, sqlName As String, hidden As String) As String

Description: This function is used to add custom nodes to all types of JSON schema.

Parameters:

- schema - The JSON metadata.
- nodeId - For example: "C004", "C005", etc.
- label - Display label for the node.
- valType - Node's value type.
- valUnit - Node's value unit.
- userType - For specific data types. The values can be "System Itemized List", "Money", "Percentage".
- sqlName - The field name in the table. DrillDown will use the field name and value as conditions in URL to pass to the detail screen. It should be used together with the configuration parameter "tableName".
- hidden - the column will not be displayed in data-list-type widget. Values are "false" or "true", by default, the value is "false".

> **Note:**

> • Before you invoke this function, you must call **createSAMWidgetSTDSchema()** or **initScriptWidgetSchema()** to generate the JSON schema.
>
> • Compared to **addScriptWidgetSchemaNode**, **addScriptWidgetSchemaNodeEx** allows you to use the "hidden" parameter to hide a column in a data-list-type widget.

Invoking sequence

1. Initialize the widget schema.
   - initScriptWidgetSchema (Non-SAM type widget)
   - createSAMWidgetSTDSchema (SAM type widget)
2. Initialize widget node properties.

   addScriptWidgetSchemaNode
3. Add node values.

   assembleNodeAttr
4. Combine node values to the schema.

   addScriptWidgetNodeValue

Example 1. List assets in stock (DataList type widget)

```
Dim lErr As Long
Dim strQuery As String
Dim hQuery As Long
Dim hRecord As Long
Dim strStockName As String
Dim strNatureName As String

Dim strWidgetSchema As String
Dim strWidgetNodeValue As String
Dim lngTotal As Long
Dim lngReserved As Long

'init widget schema

strWidgetSchema = amEvalScript("WidgetLib", "initScriptWidgetSchema", "")
```

```
'init widget node properties

strWidgetSchema = amEvalScript("WidgetLib", "addScriptWidgetSchemaNode", "",
strWidgetSchema, "C000", "Stock", "String", "", "", "Stock.Name")
strWidgetSchema = amEvalScript("WidgetLib", "addScriptWidgetSchemaNode", "",
strWidgetSchema, "C001", "Type", "String", "", "", "Asset.Model.Nature.Name")
strWidgetSchema = amEvalScript("WidgetLib", "addScriptWidgetSchemaNode", "",
strWidgetSchema, "C002", "Available", "", "", "", "")
strWidgetSchema = amEvalScript("WidgetLib", "addScriptWidgetSchemaNode", "",
strWidgetSchema, "C003", "Total", "", "", "", "")
strWidgetSchema = amEvalScript("WidgetLib", "addScriptWidgetSchemaNode", "",
strWidgetSchema, "C004", "Reserved", "", "", "", "")

strQuery = "SELECT Stock.Name, Model.Nature.Name, SUM(fQty) FROM amPortfolio WHERE
seAssignment = 1 GROUP BY Model.Nature.Name, Stock.Name"
hQuery = amQueryCreate()
lErr = amQueryExec(hQuery, strQuery)

While (lErr = 0)
    strWidgetNodeValue = ""
    strStockName = amGetFieldStrValue(hQuery, 0)
    strNatureName = amGetFieldStrValue(hQuery, 1)
    lngTotal = amGetFieldLongValue(hQuery, 2)
    lngReserved = amDbGetLong("SELECT SUM(fQty) FROM amReservation WHERE dtEnd >
GetDate() AND Assignment.Stock.Name='" & strStockName & "' AND
Assignment.Model.Nature.Name='" & strNatureName & "'")

'add node values

    strWidgetNodeValue = amEvalScript("WidgetLib", "assembleNodeAttr", "",
strWidgetNodeValue, "C000", strStockName)
    strWidgetNodeValue = amEvalScript("WidgetLib", "assembleNodeAttr", "",
strWidgetNodeValue, "C001", strNatureName)
    strWidgetNodeValue = amEvalScript("WidgetLib", "assembleNodeAttr", "",
strWidgetNodeValue, "C002", lngTotal - lngReserved)
    strWidgetNodeValue = amEvalScript("WidgetLib", "assembleNodeAttr", "",
strWidgetNodeValue, "C003", lngTotal)
    strWidgetNodeValue = amEvalScript("WidgetLib", "assembleNodeAttr", "",
strWidgetNodeValue, "C004", lngReserved)

'combine node values to schema

    strWidgetSchema = amEvalScript("WidgetLib", "addScriptWidgetNodeValue", "",
strWidgetSchema, strWidgetNodeValue)

    lErr = amQueryNext(hQuery)
Wend
lErr = amReleaseHandle(hQuery)
```

```
RetVal = strWidgetSchema
```

Example 2. Software licensing compliance of suppliers (SAM type widget)

```
Dim strComplRsltOnSupplierQry, complRsltOnSupplier As String
Dim strJSONSchema, strNodeVal As String
Dim strVendorStat, strVendorName As String
Dim lVendorComplianceVal, lVendorCounterVal As Long

'init SAM type widget schema

strJSONSchema = AMEvalScript("WidgetLib", "createSAMWidgetSTDSchema", "")

'init widget node properties

strJSONSchema = AMEvalScript("WidgetLib", "addScriptWidgetSchemaNode", "",
strJSONSchema, "C004", "Total counters", "Integer", "", "", "")
strNodeVal = ""

strComplRsltOnSupplierQry = "SELECT Brand.Company.Name, COUNT(lCountId) FROM
amSoftLicCounter WHERE (bType = 0) AND (bLicUpgrade = 0) AND (dCompliancyUpg < 0)
GROUP BY Brand.Company.Name"
complRsltOnSupplier = AmDbGetStringEx(strComplRsltOnSupplierQry, ",", ";")
If (IsEmpty(complRsltOnSupplier)) Then

'add node values

   strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C000", "Congrats!")
   strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C002", "All software counters are compliant!")
   strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C003", "GREEN")

'combine node values to schema

   strJSONSchema = AMEvalScript("WidgetLib", "addScriptWidgetNodeValue", "",
strJSONSchema, strNodeVal)
Else
   DO While (Not IsEmpty(complRsltOnSupplier))
      strNodeVal = ""
      strVendorStat = ExtractValue(complRsltOnSupplier, ";")
      strVendorName = ExtractValue(strVendorStat, ",")
```

```
      lVendorComplianceVal = CLng(strVendorStat)

      strComplRsltOnSupplierQry = "SELECT COUNT(lCountId) FROM amSoftLicCounter
WHERE (bType = 0) AND (bLicUpgrade = 0)"

'add node values

      If (IsEmpty(strVendorName)) Then
         strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C000", "N/A")
         strComplRsltOnSupplierQry = strComplRsltOnSupplierQry + " AND (lBrandId=0
OR Brand.lCompany=0)"
      Else
         strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C000", strVendorName)
         strComplRsltOnSupplierQry = strComplRsltOnSupplierQry + " AND
Brand.Company.Name='" + strVendorName + "'"
      End If
      lVendorCounterVal = AmDbGetLong(strComplRsltOnSupplierQry)
      strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C004", lVendorCounterVal)

      If (lVendorComplianceVal > 0) Then
         strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C001", lVendorComplianceVal)
         strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C002", "non-compliant counters")
         strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C003", "RED")
      Else
         strNodeVal = AMEvalScript("WidgetLib", "assembleNodeAttr", "", strNodeVal,
"C003", "GREEN")
      End If

'combine node values to schema

      strJSONSchema = AMEvalScript("WidgetLib", "addScriptWidgetNodeValue", "",
strJSONSchema, strNodeVal)
   Loop
End If

RetVal = strJSONSchema
```

Personalizing the homepage

When the first time you log on to the Web client, the layout of the homepage is defined by the homepage template created for your user role. The homepage is quite intuitive and you can also simply personalize it by:

- Adding or removing a widget by clicking the giant "+" button at the bottom of the homepage or the ✕ button.
- Expanding or collapsing a widget by clicking the button or the button.
- Dragging and dropping the widgets.
- Changing the parameters of the widget by clicking the button.
- Exporting a widget as a .png file (chart) or a .csv file (data list) by clicking the button.
- Refreshing a widget by clicking the button.
- Maximizing a widget by clicking the button.

After you make any changes to your homepage, the changes are saved automatically.

User profiles

The out-of-box data of Asset Manager contains the **Homepage widgets management** and **Homepage templates management** profiles. A user with the **Homepage widgets management** profile can create homepage widgets. Likewise, a user with the **Homepage templates management** profile can create homepage templates for user roles.

If you do not have the out-of-box data imported in your environment, we recommend that you create similar user profiles as described in this section.

Homepage widgets management

The **Homepage widgets management** user profile consist of the following user rights and functional rights.

User rights

- **Homepage templates configuration (amHomepageTemplate)** - Read
- **InToolReport (amInToolReport)** - Full access
- **MyFavorites (amMyFavorites)** - Full access
- **ScriptReport (amScriptReport)** - Read
- **ScriptReportLibrary (amScriptReportLibrary)** - Read
- **UIWidget (amUIWidget)** - Full access
- **UIWidget Instance (amUIWidgetInstance)** - Full access

Functional rights

- **Administration** > **Full** > **Home page widgets (HomepageWidgets)** - Granted

Homepage templates management

The **Homepage templates management** user profile consist of the following user rights and functional rights.

User rights

- **Homepage templates configuration (amHomepageTemplate)** - Full
- **InToolReport (amInToolReport)** - Read
- **MyFavorites (amMyFavorites)** - Full access
- **ScriptReport (amScriptReport)** - Read
- **ScriptReportLibrary (amScriptReportLibrary)** - Read
- **UIWidget (amUIWidget)** - Full access
- **UIWidget Instance (amUIWidgetInstance)** - Full access

Functional rights

- **Administration** > **Full** > **Home page templates (HomepageTemplates)** - Granted

How to create a homepage template

Asset Manager allows every user to personalize the homepage of the Web client. In addition, a user with the **Homepage templates management** profile can create homepage templates for certain user roles. By default, when a user logs on to the Web client, the template defined for his/her user role is displayed as the homepage.

> **Note:** For more details of the **Homepage templates management** profile, see "User profiles" on page 184.

To create a homepage template, follow these steps:

1. On the navigation bar, expand **Administration**, and then click **Home page templates**.
2. Select a user role.
3. Create the template for the selected user role by using the method described in "Personalizing the homepage" on page 184.
4. The **Status** column will display a green check and the template is saved automatically.

> **Note:** If a user makes any changes to his/her homepage widgets, the subsequent changes to the homepage template no longer affects that user's homepage.

Importing/exporting homepage widgets and templates

Because of schema changes, in Asset Manager 9.61, there exist two types of widgets, 9.41-format and 9.61-format. You must use different tools (scripts) to import/export them respectively.

Importing/exporting homepage widgets

If you upgrade Asset Manager from version 9.41 to 9.61, you need to notice the following facts.

- In Asset Manager 9.61, a widget that is created in Asset Manager 9.41 is still in 9.41-format. To import/export this type of widgets, you must run the 9.41 import/export scripts.
- In Asset Manager 9.61, if you modify a widget that is created in Asset Manager 9.41, the widget is still in 9.41-format. To import/export this type of widgets, you must run the 9.41 import/export scripts.
- In Asset Manager 9.61, if you duplicate a widget that is created in Asset Manager 9.41, the new widget is in 9.50-format. To import/export this type of widgets, you must run the 9.50 import/export scripts.

In Asset Manager 9.61, you can run the following scripts to import and export 9.41-format widgets.

- **Import**
  - <AM installation directory>\datakit\bestprac\portfolio\IntoolReport_9.41.scr
  - <AM installation directory>\datakit\bestprac\procurement\IntoolReport_9.41.scr
  - <AM installation directory>\datakit\bestprac\sam\IntoolReport_9.41.scr
  - <AM installation directory>\datakit\bestprac\finance\IntoolReport_9.41.scr
  - <AM installation directory>\datakit\bestprac\helpdesk\IntoolReport_9.41.scr
- **Export**
  - <AM installation directory>\datakit\bestprac\portfolio\IntoolReport_9.41.scx
  - <AM installation directory>\datakit\bestprac\procurement\IntoolReport_9.41.scx
  - <AM installation directory>\datakit\bestprac\sam\IntoolReport_9.41.scx
  - <AM installation directory>\datakit\bestprac\finance\IntoolReport_9.41.scx
  - <AM installation directory>\datakit\bestprac\helpdesk\IntoolReport_9.41.scx

In Asset Manager 9.61, you can run the following scripts to import and export widgets that are created using Asset Manager 9.61.

- **Import**
  - <AM installation directory>\datakit\bestprac\portfolio\widget.scr
  - <AM installation directory>\datakit\bestprac\procurement\widget.scr
  - <AM installation directory>\datakit\bestprac\sam\widget.scr
  - <AM installation directory>\datakit\bestprac\finance\widget.scr
  - <AM installation directory>\datakit\bestprac\helpdesk\widget.scr

- **Export**
  - <AM installation directory>\datakit\bestprac\portfolio\widget.scx
  - <AM installation directory>\datakit\bestprac\procurement\widget.scx
  - <AM installation directory>\datakit\bestprac\sam\widget.scx
  - <AM installation directory>\datakit\bestprac\finance\widget.scx
  - <AM installation directory>\datakit\bestprac\helpdesk\widget.scx

> **Note:** The reconciliation keys of 9.41-format widgets consist of both the **widgetName** field from the **amUIWidget** table and the **Name** field from the **amInToolReport** table. If either the **widgetName** field or the **Name** field is changed, the import script will insert a new widget into the database even though the widget already exists. If you run into this error, you need to manually check the database and make sure these two fields have the same value.

Importing/exporting homepage templates

If you upgrade Asset Manager from version 9.41 to 9.61, you need to notice the following facts.

- In Asset Manager 9.61, a homepage template that is created in Asset Manager 9.41 is still in 9.41-format. To import/export this type of widgets, you must run the 9.41 import/export scripts.
- In Asset Manager 9.61, if you modify a homepage template that is created in Asset Manager 9.41, the format of the template becomes 9.61. To import/export this type of widgets, you must run the 9.61 import/export scripts.

In Asset Manager 9.61, you can run the following scripts to import and export 9.41-format homepage templates.

- **Import**
  - To import the widgets that are added to the template.

    <AM installation directory>\datakit\bestprac\homepage_template\HomepageTemplate_part1_9.41.scr
  - To import the templates.

    <AM installation directory>\datakit\bestprac\homepage_template\HomepageTemplate_part2_9.41.scr
- **Export**

- To export the widgets that are added to the template.

  <AM installation directory>\datakit\bestprac\homepage_template\HomepageTemplate_part1_9.41.scx

- To export the templates.

  <AM installation directory>\datakit\bestprac\homepage_template\HomepageTemplate_part2_9.41.scx

In Asset Manager 9.61, you can run the following scripts to import and export 9.50-format homepage templates.

- **Import**
  - To import the widgets that are added to the template.

    <AM installation directory>\datakit\bestprac\homepage_template\widget.scr

  - To import the templates.

    <AM installation directory>\datakit\bestprac\homepage_template\HomepageTemplate.scr

- **Export**
  - To export the widgets that are added to the template.

    <AM installation directory>\datakit\bestprac\homepage_template\widget.scx

  - To export the templates.

    <AM installation directory>\datakit\bestprac\homepage_template\HomepageTemplate.scx

Business home pages

This section introduces business home pages.

This section includes:

What is a business home page? 190
How business home pages work 190
How to create or modify a business home page 191
Recommendations when creating business home pages 191
HTML syntax for useful links 191
Specifying default home pages and home pages based on user roles 193

Deleting a business home page 195

> **Note:** Business home pages can only be viewed in the Asset Manager Web client.
>
> They can be created using the Windows or Web client.

What is a business home page?

A business home page provides a summary screen for a given functional domain and connection role (See Specifying default home pages and home pages based on user roles).

When you click a domain node on the navigation bar a business home page is displayed (for example, **Finance**).

It has different sections:

- Statistics
- A set of useful links in HTML format: open a screen, start a wizard, open a Web site, and so on.
- Actions
- Reports

How business home pages work

- Business home page contents may be viewed by multiple users, but they are only displayed if the user's role authorizes it.
  If a user has several connection roles, they need to select the appropriate role (**Administration/ User actions/ Select your Asset Manager user role** link on the navigation bar).
- If one of the sections of the business home page does not contain any data, then the section is not displayed.
- Asset Manager considers that a report is available via SAP BusinessObjects Enterprise if:
  - The SAP BusinessObjects Enterprise was declared via the **Edit the URL address of the application servers...** (BstBackEndOpt) wizard (**Administration/ User actions/ Edit the URL address of the application servers...** link on the navigation bar).
  - And the report's **File** (FileName) field contains a **numerical value** corresponding to its identifier in the SAP BusinessObjects Enterprise server.

For more information about SAP Crystal Reports, refer to the **Advanced use** guide, chapter **SAP Crystal Reports**.

How to create or modify a business home page

1. Display the business home pages (amFDView): **Administration/ Business home pages** link on the navigation bar.
2. Create a new record by populating the relevant fields and adding the desired items on the different tabs, or select the record to modify and make the desired changes.
3. Save your changes.

The business home page that you have created is now available. In the Web client click the node specified in the **Domain** (Domain) field.

Recommendations when creating business home pages

We recommend that you not use more than two statistics per page (however, the maximum number is six).

You can set the maximum number in Asset Manager Web Tier's **package.properties** file by modifying the **BusinessHomePages.Items.Count.Maximum** parameter.

See Modifying the maximum number of statistics to display in business home pages.

HTML syntax for useful links

Useful links are entered in HTML. The following syntax rules must be used:

Unfiltered Asset Manager lists 192
Asset Manager lists filtered by query 192
Asset Manager lists filtered on an indexed field 192
Asset Manager Wizards 193

External Web pages 193

Unfiltered Asset Manager lists

```
<a href="/AssetManager/<SQL name of the top-level domain in the tree structure to which the list is attached>/<SQL name of the screen without the 'am' prefix>/list.jsf"><Label of the link to display on the business home page></a>
```

For example:

```
<a href="/AssetManager/IT/CPUComputer/list.jsf">Computers</a>
```

> **Note:** Example of **am** prefix removal: Instead of typing **amPOrder** for the purchase orders screen, type only **POrder** instead.

Asset Manager lists filtered by query

```
<a href="/AssetManager/<SQL name of the top-level domain in the tree structure to which the list is attached>/<SQL name of the screen without the 'am' prefix>/list.jsf?filterName=<SQL name of the query>"><Label of the link to display on the business home page></a>
```

For example:

```
<a href="/AssetManager/Core/EmplDept/list.jsf?filterName=BstEmplWithProfile">Employees with a user profile</a>
```

Asset Manager lists filtered on an indexed field

```
<a href="/AssetManager/<SQL name of the top-level domain in the tree structure to which the list is attached>/<SQL name of the screen without the 'am' prefix>/list.jsf?filterName=<SQL name of the field>&<SQL name of the field>=<Value to apply"><Label of the link to display on the business home page></a>
```

For example:

```
<a href="/AssetManager/Core/Location/list.jsf&filterName=BarCode&BarCode=M%25">Locations where the bar code is equal to M%25</a>
```

Asset Manager Wizards

```
<a href="/AssetManager/<SQL name of the top-level domain in the tree structure to
which the wizard is attached>/<SQL name of the wizard>/wizard.jsf??filterName=<SQL
name of the filter>"><Label of the link to display on the business home page></a>
```

For example:

```
<a href="/AssetManager/Procurement/sysProcRequestAddModelnc/
wizard.jsf?">Create a new request...</a>
```

> **Note:** To reference Asset Manager objects, you do not need to enter the complete URL address. You are not required to enter the domain name (for example: **http://MyServerAssetManager.com:8090/**).

External Web pages

To reference external Web pages, enter the complete URL address to the pages.

Specifying default home pages and home pages based on user roles

The business home page of a given domain is used for all of its sub-domains except if the sub-domain has its own home page.

You can also tailor home pages to specific user roles. In this case, you will need to create a home page per role. To do this, specify the desired user role in the **User role** (MasterProfile) field.

By default, if you do not specify a user role, the home page will be displayed for all users.

For example, let's suppose we have:

- **User U** whose user role is **CIO**.
- **Standard user S**.

And they are accessing the following sections of the navigation bar:

- Portfolio management
  - IT
    - IT equipment

The following table details the different business home pages that will be displayed depending on the connected user:

Function domain (or sub-domain)	Business home pages that are created	Standard user S clicks the domain name (or sub-domain). The following business home page is displayed:	User U clicks the domain name (or sub-domain). The following business home page is displayed:
Portfolio management	A single page is created without specifying the user role: **page A**	page A	page A
IT	Two pages are created: • **Page B1** without specifying a user role. • **Page B2** when specifying the **CIO** user role.	page B1	page B2
IT equipment	Case 1: No page is created	page B1	page B2
	Case 2: A single page is created without specifying the user role: **page C**	page C	page C
	Case 3: A single page is created when specifying the **CIO** user role: **page D**	page B1	page D
	Case 4: Two pages are created: • **Page E1**	page E2	page E2

Function domain (or sub-domain)	Business home pages that are created	Standard user S clicks the domain name (or sub-domain). The following business home page is displayed:	User U clicks the domain name (or sub-domain). The following business home page is displayed:
	without specifying a user role. • **Page E2** when specifying the **CIO** user role.		

Deleting a business home page

To delete a business home page:

1. Display the business home pages (amFDView): **Administration/ Business home pages** link on the navigation bar.
2. Select the record to delete.
3. Click the **Delete** button.

Send documentation feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Tailoring (Asset Manager 9.61)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to ovdoc-ITSM@hpe.com.

We appreciate your feedback!