


Working with the Default EUI Templates

This topic provides instructions on how to control which tables are displayed on the [End User Interface](#) home page and menu bar, add new tables to these sections, and some code samples to set up these areas in the templates.

 The end user interface is controlled by records in the EUI Templates table. These records contain the HTML for the pages that are displayed and control the functionality and appearance of the EUI. For more information on getting started with the EUI, please review Unit 19 of the online Administrator Training at [unit-19-eui.pdf](#).

The EUI uses macros and methods which are interpreted by the Apache [Velocity](#) template engine. For more information on the macros and methods used in an Agiloft EUI, please refer to the Macros Reference section of the EUI Tips and Reference Guide at <http://www.agiloft.com/documentation/eui-reference-guide.pdf>.

Home Page Layouts

Near the top of the `translation.properties` file are three variables that define the home page template files. The variable `global.home.file` defines which of two predefined home pages to use: the default value of `home.htm`, or `home-1column.htm`:

```
global.home.file=home-1column.htm
#The default home page is home.htm, or you can use home-1column.htm by replacing
this variable
global.home.one.table.file=home-1table.htm
global.search.file=home-multiple-search.htm
```

In addition to these default home pages, the code automatically substitutes a third home page if the user who logs in only has access to one table, since the other home page layouts are not optimized for displaying a single table. This third home page is defined by the variable `global.home.one.table.file` and by default it is set to the `home-1table.htm` template.

The home page code defines which tables are shown to which users, the ordering of tables, and other formatting. Of course, you may create your own version of a home page if you don't wish to use our predefined versions.

While some page components are static, such as the Logout link, other links and headings are controlled by a combination of user permissions and variables defined in `translations.properties`.

These conditions include the values of the `show.[table]` variable, as well as the user's specific permissions, returned by the `$ewPermission.table` method.

Making Changes or Substituting a New Home Page

To make changes to the style or layout of your end user interface home page, copy and edit one of the default home page records, or create a new template record. Then, update the `global.home` and `global.home.one.table` variables in `translation.properties` to point to your new page.

Changing the Order of Tables in the Home Page

There is a block of code for each table included in the home page; the order of these sections determines the order in which tables are displayed to users.

Therefore, to change the order of the tables, you can cut and paste one table's section above or below another one. Just be careful to include all of the section related to that table: each section should start with an `#if` and end with an `#end` - watch out for nested if-end conditions.

Specifically, each block of table code begins with:

```
#if ($ewText.get("show.[table]")=="yes"
```

Please review the [Home Page Table Block](#) example code from the home.htm template, the default home page, to see how the permission-based conditions are used.

We recommend first making a copy of the template text before making significant changes.

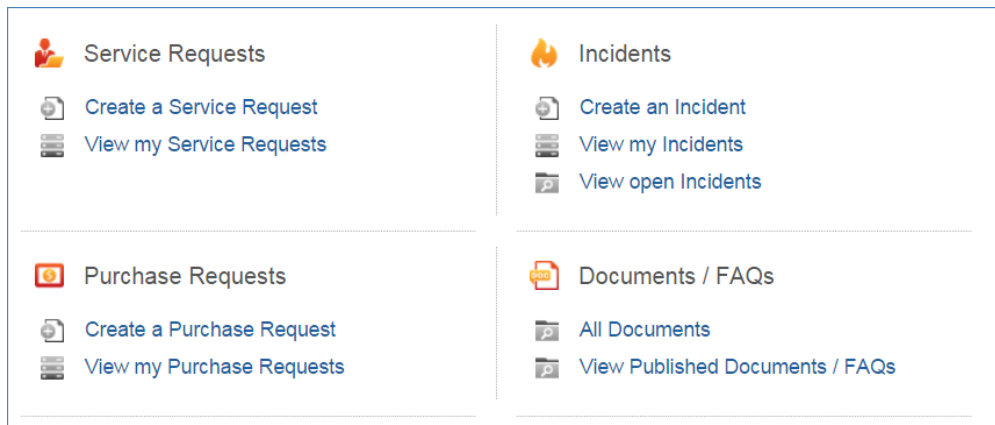
Default Template Layouts

The default home page layouts are:

home.htm—Default layout













The default `home.htm` template provides a dynamic two-column display; the system dynamically calculates the layout to list the available tables in order from left-to-right and top-down.

Available links to create or view items appear under each table heading, as pictured here.



home-1column.htm—Alternative layout

The alternative home page `home-1column.htm` arranges each table on its own row with the links in a second column to the right:





 Service Requests	 Create a Service Request  View my Service Requests
<hr/>	
 Incidents	 Create an Incident  View my Incidents  View open Incidents
<hr/>	
 Purchase Requests	 Create a Purchase Request  View my Purchase Requests
<hr/>	
 Documents / FAQs	 View Published Documents / FAQs

home-1table.htm—Default one-table layout

The `home-1table.htm` template uses a similar layout to the `home-1column.htm` template, above. The primary differences are larger icons and text in `home-1table.htm`, and an additional block of text added to fill space on the home page.

Welcome to the Agiloft Custom Portal

You are on the Custom Portal Homepage. From here you can create new records, View or edit existing records, and depending on your permissions and your company setup, you may be able to browse frequently asked questions by clicking FAQs and view or update your user account information using My Profile.

 Support Cases	 Create a Support Case  View my Support Cases  FAQs
---	--

Navigation Menu Structure

The navigation menu and its contents are defined in the `menu.htm` EUI template. Please refer to the [Code Samples](#) section for an example of that template's text for the section controlling the New menu. You can see that it uses permission controls very similar to those used in the home page.



The menu template is used to count the number of tables available to a user and if the table count in the menu results in a total value of 1, then the home page defined as the single table home page is displayed to the user.

Changing which Menu Items a User Sees

Just as in the home page, the menu items are controlled by a combination of the `show.[table]` variable and user permissions, so if you want to hide the item from all users, the easy way is to update the translation properties file to change the `show.[table]` to no. If it should be shown to only users with specific permissions, then the approach is to update the relevant `#if` condition to use the appropriate permissions.

Adding a New Table to the EUI

If you want to add a table to the EUI that is not included in the default setup, you must update the EUI templates to provide this access.

For this example, we'll suppose you have a custom table called Invoices and you would like to add functions to the EUI such as "Create an Invoice" and "My Invoices," a link to view invoices for end users.

Below are the basic recommended steps for adding a new table:

1. Create new template pages that will be launched from the home page. We recommend copying a similar EUI template record and then changing the page-specific information such as the logical table name, macro parameters, and text variables. For instance, to create a page to display the Invoices table, copy the `my_supportcases.html` page into a new template called `my_invoices.html`; to create a "Create an Invoice" page, copy the `new_supportcase.html` template.
 - a. Edit the new template pages to update the logical table name, change the default view, add any default saved searches, and so on.
 - b. If the page you copy from includes a search block, you will need to create a corresponding subsidiary search page for the new table, Invoices.
 - c. Change the variables within the `$ewText.get` macro. Note down the variables you create and the text strings they represent. These variables and values must be added to `translation.properties` in the following step.

2. In the translation.properties file (and its variants for multilingual KBs):

a. Add the line `show.invoices=yes`

This will let you turn the Invoices table on or off quickly and allows you to use that variable in the new section of table code. For instance, in `home.htm` or `menu.htm` where you previously had `#if ($ewText.get("show.support.cases")==="yes",` replace `show.support.cases` with `show.invoices`.

b. Add any text variables created for the subsidiary pages in the previous step.

c. Create new text variables for the home page and menu: the table title, create link, view links, menu tabs, etc. Use the set of variable codes from an existing table and create new variables using a similar syntax. Since we are copying from Support Cases in this example, copy the text variables (in the left column) and create new variables for the Invoices table in a similar syntax (shown in the right-hand column).

Support Case Text	Invoice Text
<code>home.sc.h3=Support Cases</code>	<code>home.invoice.h3=Support Cases</code>
<code>home.sc.my.link=View My Support Cases</code>	<code>home.invoice.my.link=View My Invoices</code>
<code>home.sc.new.link=Create a Support Case</code>	<code>home.invoice.new.link=Create an Invoice</code>
<code>menu.new.sc=Support Case</code>	<code>menu.new.invoice=Invoice</code>
<code>menu.view.sc.my=My Support Cases</code>	<code>menu.view.invoice.my=My Invoices</code>

3. Edit the default home page(s) you are using.

a. Copy an existing table's complete section of code, e.g. the Support Case section, and paste it to create a new section of the home page template. Paste the section into the appropriate position so the tables are ordered as you prefer.

b. Modify the copied block of code by replacing variables and parameters with new values associated with the table you want to display (Invoices). For example:

i. Update the `#ew_forward` macro to point to the new template pages created in the previous step.

ii. Update the `$ewText.get` statements to use the variables created in the previous step.

iii. Update the logical table name (i.e., case) wherever it appeared.

4. To add links from the menu, add new sections to the `menu.htm` template in the appropriate places (e.g., under the New and View sections). Copy an existing block of code and change the table name, text variables, and criteria as needed.

5. Of course, you must set group permissions to give users the appropriate access to the Invoices table and its records.

If the user has create permissions for support cases, then display the New Support Case link.

```
#if ($ewPermission.table("case","view_own") || $ewPermission.table("case","v_own_with_ssearch"))  
  
<a href='#ew_forward("my_supportcases.html")'>$ewText.get('home.sc.my.link')</a><br />  
</a>  
</div>  
</div>  
</td>
```

If the user has view own or view own with saved search record permissions, then show the My Support Cases link.

```
#if ($ewText.get("show.sc.faqs")==="yes" && ($ewPermission.table("case","view faq") || $ewPermission.table("case","view faq_with_ss")))  
  
<a href='#ew_forward("supportcases-faq.html")'>$ewText.get('home.sc.faq.link')</a>  
</div>  
</div>  
</td>
```

If the show support case FAQs is yes and the user has View FAQs or view FAQs with saved search permissions on the Support Case table, show the FAQs link.

```
</div>  
</div>  
</td>
```

```
#if ($rowcount%2==0)  
</tr>  
<tr>  
<td class="h-line"></td>  
<td></td>  
<td class="h-line"></td>  
</tr>  
<tr>  
<td class="v-line"></td>  
</tr>  
</td>
```

This defines whether it is the first column or second column and inserts the appropriate vertical or horizontal lines.

Add the Table to Global Search

The following code sample will add the new table to the global search area for the EUI. In this example the Invoice table will be added to the global search.

1. In `home-multiple-search.htm`, add the following:

```
#if (($ewText.get("show.invoice")=="yes")&&($ewPermission.table("invoices","view_own")))
<input name="tables" type="checkbox" value="invoice" id="invoice"> <label for="task">$ewText.get("home.search.invoice")</label>
<br>
#end
```

2. In `multiple-search-result.html` add the following:

```
#if ($queryTables.contains('invoice') )
<div class="main_frames">
#ew_table("invoices" "" $defaultsearch ""
"showToolBar=false&showNavigation=true&showfastsearch=false&showSearchCriteria=f
```

And

```
#if ($queryTables.contains('invoice'))
#ew_query("invoices" "my_table10" "queryId"
"showToolBar=false&showNavigation=false&showfastsearch=false")
#end
```

Navigation Menu

This section contains samples to add items in the navigation menu.

New Menu

Below is the section of `menu.htm` that creates the New menu drop-down in the EUI.

New

View

Helpdesk Case

Support Case

Contract

Document

```

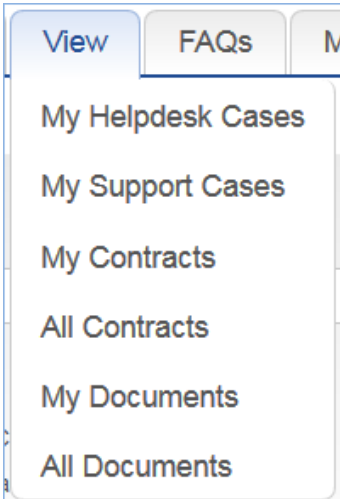
#if ($ewText.get("show.menu.new")==="yes")
<li id="new" class="submenu"><a>$ewText.get("menu.new")</a>
<ul>
#if
($ewText.get("show.helpdesk.cases")==="yes" && $ewPermission.table("helpdesk_case", "
create"))
<li><a href='#ew_forward("new_helpdeskcase.html")'>$ewText.get("menu.new.hc")</a><
/li>
#end
#if
($ewText.get("show.support.cases")==="yes" && $ewPermission.table("case", "create"))
<li><a href='#ew_forward("new_supportcase.html")'>$ewText.get("menu.new.sc")</a><
/li>
#end
#if
($ewText.get("show.change.requests")==="yes" && $ewPermission.table
("change_request", "create"))
<li><a href='#ew_forward("new_changerequest.html")'>$ewText.get("menu.new.cr")</a><
/li>
#end
#if ($ewText.get("show.contracts")==="yes" && $ewPermission.table("contract", "
create"))
<li><a href='#ew_forward("new_contract.html")'>$ewText.get("menu.new.con")</a></li>
#end
#if ($ewText.get("show.documents")==="yes" && $ewText.get("show.document.creation")
=="yes" && $ewPermission.table("document", "create"))
<li><a
href='#ew_forward("new_document.html")'>$ewText.get("menu.new.doc")</a></li>
#end
#if ($ewText.get("show.tasks")==="yes" && $ewPermission.table("activities", "create"))
<li><a href='#ew_forward("new_task.html")'>$ewText.get("menu.new.task")</a></li>
#end
#if ($ewText.get("show.repairs")==="yes" && $ewPermission.table("repair_request", "
create"))
<li><a href='#ew_forward("new_repair.html")'>$ewText.get("menu.new.rep")</a></li>
#end

```

Essentially the items are controlled by the show.[table]=yes variable and the create permissions for a particular user.

View Menu

Below is the section of menu.htm that creates the View menu drop-down in the EUI.



```
#if($ewText.get("show.menu.view")==="yes")
<li id="view" class="submenu"><a>$ewText.get("menu.view")</a>
<ul>
#if ($ewText.get("show.helpdesk.cases")==="yes" && ($ewPermission.table
("helpdesk_case","view_own") || $ewPermission.table("helpdesk_case","
v_own_with_ssearch")))
#set($htable=1)
#set($tablecount=$tablecount+1)
<li><a href='#ew_forward("my_helpdeskcases.html")'>$ewText.get("menu.view.hc.my")<
/a></li>
#end
#if
($ewText.get("show.support.cases")==="yes" && ($ewPermission.table("case","
view_own") || $ewPermission.table("case","v_own_with_ssearch")))
#set($sctable=1)
#set($tablecount=$tablecount+1)
<li><a href='#ew_forward("my_supportcases.html")'>$ewText.get("menu.view.sc.my")<
/a></li>
#end
#if
($ewText.get("show.change.requests")==="yes" && ($ewPermission.table
("change_request","view_own") || $ewPermission.table("change_request","
v_own_with_ssearch")))
#set($tablecount=$tablecount+1)
<li><a href='#ew_forward("my_changerequests.html")'>$ewText.get("menu.view.cr.my")<
/a></li>
#end
#if ($ewText.get("show.contracts")==="yes" && ($ewPermission.table("contract","
view_own") || $ewPermission.table("contract","v_own_with_ssearch")))
#set($tablecount=$tablecount+1)
<li><a href='#ew_forward("my_contracts.html")'>$ewText.get("menu.view.con.my")</a><
/li>
<li><a href='#ew_forward("all_contracts.html")'>$ewText.get("menu.view.con.all")<
/a></li>
#end
```

```
#if ($ewText.get("show.documents")==="yes" && ($ewPermission.table("document", "view_own") || $ewPermission.table("document", "v_own_with_ssearch")))
#set($tablecount=$tablecount+1)
#set($doctable=1)
<li><a href='#ew_forward("my_documents.html")'>$ewText.get("menu.view.doc.my")</a></li>
<li><a href='#ew_forward("all_documents.html")'>$ewText.get("menu.view.doc.all")</a></li>
#end
#if ($ewText.get("show.tasks")==="yes" && ($ewPermission.table("activities", "view_own") || $ewPermission.table("activities", "v_own_with_ssearch")))
#set($tablecount=$tablecount+1)
<li><a href='#ew_forward("my_opentasks.html")'>$ewText.get("menu.view.opentask.my")</a></li>
#end
#if ($ewText.get("show.repairs")==="yes" && ($ewPermission.table("repair_request", "view_own") || $ewPermission.table("repair_request", "v_own_with_ssearch")))
#set($tablecount=$tablecount+1)
<li><a href='#ew_forward("my_repairs.html")'>$ewText.get("menu.view.rep.my")</a></li>
#end
```