



Customer Relationship
Management

CRM Mobile Administrator Guide

Version 14.1



Notices

Copyright © 2004–2022. Aurea Software, Inc. (“Aurea”). All Rights Reserved. These materials and all Aurea products are copyrighted and all rights are reserved by Aurea.

This document is proprietary and confidential to Aurea and is available only under a valid non-disclosure agreement. No part of this document may be disclosed in any manner to a third party without the prior written consent of Aurea. The information in these materials is for informational purposes only and Aurea assumes no responsibility for any errors that may appear therein. Aurea reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of Aurea to notify any person of such revisions or changes.

You are hereby placed on notice that the software, its related technology and services may be covered by one or more United States (“US”) and non-US patents. A listing that associates patented and patent-pending products included in the software, software updates, their related technology and services with one or more patent numbers is available for you and the general public’s access at <https://markings.ip-dynamics.ai/esw/> (the “Patent Notice”) without charge. The association of products-to-patent numbers at the Patent Notice may not be an exclusive listing of associations, and other unlisted patents or pending patents may also be associated with the products. Likewise, the patents or pending patents may also be associated with unlisted products. You agree to regularly review the products-to-patent number(s) association at the Patent Notice to check for updates.

Aurea and Aurea Software are registered trademarks of Aurea Software, Inc. in the United States and/or other countries. Additional Aurea trademarks, including registered trademarks, are available at: <https://www.aurea.com/legal/trademarks/>. Jive is a registered trademark of Jive Software, Inc. in the United States and/or other countries. Additional Jive trademarks, including registered trademarks, are available at: <https://www.jivesoftware.com/legal/>.

Table of Contents

Preface	5
About this documentation.....	5
Notation conventions.....	5
Aurea global support.....	6
Chapter 1: CRM.mobile overview	7
Access rights.....	8
System requirements & supported devices.....	8
Client Log.....	9
Chapter 2: Setting Up CRM.mobile	10
CRM.designer.....	11
Users configuration.....	11
Web configuration.....	13
Chapter 3: Using CRM.mobile	17
Info Area Customizations.....	17
Field Groups.....	17
Expand configuration.....	31
Search&List Configuration.....	35
CRM.mobile-Specific Configurations.....	37
Table captions.....	38
Info Area Labels.....	38
Navigation customizations.....	39
Default action.....	39
Menus.....	40
Buttons.....	45
Headers.....	50
Global Search.....	53
Action templates.....	55
EditView-NewView.....	56
RecordListView.....	57
Tiles.....	58
Tile-Calendar.....	59
Tile-CalendarAgenda.....	60
Tile-calendarMonth.....	62
Tile-RecentlyUsed.....	63

Tile-Record.....	64
Tile-Search.....	65
CalendarView.....	66
Filters.....	68
Defining default values.....	68
Languages and Texts.....	71

Preface

For details, see the following topics:

- [About this documentation](#)
- [Notation conventions](#)
- [Aurea global support](#)

About this documentation

This guide is part of the documentation set for Aurea CRM.

Notation conventions

This document uses the following notation conventions:

Convention	Meaning
Fixed-width	Fixed-width font indicates code, path names, file names, environment variable names, parameter names, command names, machine names, URLs.
Fixed-width	Fixed-width font is used to indicate user input or to emphasize certain lines of code.
<i>Fixed-width</i>	<i>Fixed-width</i> font indicates a placeholder for which you must supply a value.
Sans serif	sans serif typeface indicates the names of graphic user interface elements such as dialog boxes, buttons, and fields.
<i>serif</i>	In text, <i>serif</i> typeface indicates the first use of an important term. The term is defined in the glossary.
Underlined	Underlined text in command lines and parameter descriptions indicate that you only have to enter the underlined part of the command or parameter name. For example, if you use the <u>-LOGFILE</u> parameter in a command, you only need to enter <code>-LOGF</code> .
[]	Brackets enclose optional arguments.
{ a b c }	Braces enclose two or more items. You can specify only one of the enclosed items. Vertical bars represent OR separators. For example, you can specify a or b or c.

Convention	Meaning
...	Three consecutive periods indicate that you can repeat the immediately previous item. In code examples, they can be horizontal or vertical to indicate omissions.
Menu > Choice	An angle bracket between two menu items indicates that you should choose an item from a menu. For example, the notation File > > Exit means: "Open the File menu and choose Exit ."
>>	Links to related information in other chapters or documents are indicated using the >> symbol.

Aurea global support

If you encounter a problem while using an Aurea product or require assistance with downloading the software or upgrading a product release, please open a ticket on [Aurea Support Central](#). Preferably, search the articles on the [Aurea Knowledge Base](#) for solutions to your issues before opening a ticket.

Information about the support organization is available on Support Central. The product documentation is available at <https://help.aurea.com/crm/#>.

For information about purchasing an upgrade or professional services, contact your account executive. If you do not know who your account executive is, or for other queries, [contact us](#) through our [website](#).

1

CRM.mobile overview

CRM.mobile is the mobile version of the CRM web, enabling users to view and edit the contents of the CRM database on their mobile devices.

CRM.mobile supports all available verticals. It is configured via CRM.designer using update.tablet as the default configuration. It requires online access to the update.CRM and CRM.designer databases.

Note: The CRM.mobile shares its default configuration update.tablet with CRM.pad this implies that the changes to configuration units (e.g. a Details control or a placeholder definition) may affect the appearance and/or functionality of both CRM.mobile and CRM.pad.

This documentation covers the following:

- Setup of CRM.mobile. For further details, see [Setting up CRM.mobile](#)
- Customization of CRM.mobile using CRM.designer. For further details, see [Configuring CRM.mobile using CRM.designer](#).

Additional information on CRM.mobile-specific configuration units is provided in the following documentation sets:

- For additional information on customizing CRM.mobile, see [Advanced designer customization](#).
- For details on installing CRM.web and CRM.designer, see [CRM web technical & installation guide](#).
- For general information on using CRM.designer and customizing web-based CRM applications, see [CRM web Administrator Guide](#).
- The administration and configuration of core functions are described in the CRM.Core Administrator Guide, see [CRM.core administrator guide](#).

Access rights

CRM.mobile provides users with different access rights defined in the update.CRM Win rights module.

Please note the following:

- **Viewing records (details/list):** Fields with **Deny Access** are displayed as empty.
- **New/Edit mode:** If a user has insufficient rights to edit a field, the field is displayed read-only. If a user has no update rights for an entire info area, the entire edit mask is read-only.

Fields defined as read-only in the data model or via the `readonly` field attribute, are read-only as well. For further details, see [Readonly](#).

- Conditional rights where the condition is defined for a record's parent are currently not supported in CRM.mobile. For example, a rights definition where a user can only create MA records for companies from a certain country completely locks the Edit mask of MA.
- Field validation is currently only performed by the core.
- Non-interactive triggers from the rights definition are executed.

Note: Plausibility check triggers are not supported by CRM.mobile.

To view CRM.mobile's client log, start CRM.mobile with the URL parameter `?debug`. For further details, see [Client log](#).

System requirements & supported devices

CRM.mobile is supported by many devices. Before installing it, learn the system requirements of CRM.mobile.

Here's a list of supported devices and system dependencies.

Web server	Notes
Internet Information Services 7.5	Yes
Internet Information Services 8.0	Yes

Client Operating System for tablet	Notes
iOS	6.x, 7.1.x, 8.1x
Android	4.2, 4.4, 4.5

Client Operating System for tablet	Notes
Windows	8, 8.1
Blackberry	10.2

Client Log

Learn to view CRM.mobile's client log.

To view CRM.mobile's client log start CRM.mobile with the URL parameter `?debug`. For details on the client log, see [Client Log](#) in the Aurea.CRM Web Administrator Guide.

2

Setting Up CRM.mobile

CRM.mobile is a highly customizable platform for mobile devices. It can be configured by CRM.designer.

CRM.mobile comes as a server plug-in that is installed as part of CRM.web. The plug-in is enabled by default via Aurea.CRM's web's `settings.xml` file. Therefore, no additional setup routine needs to be executed.

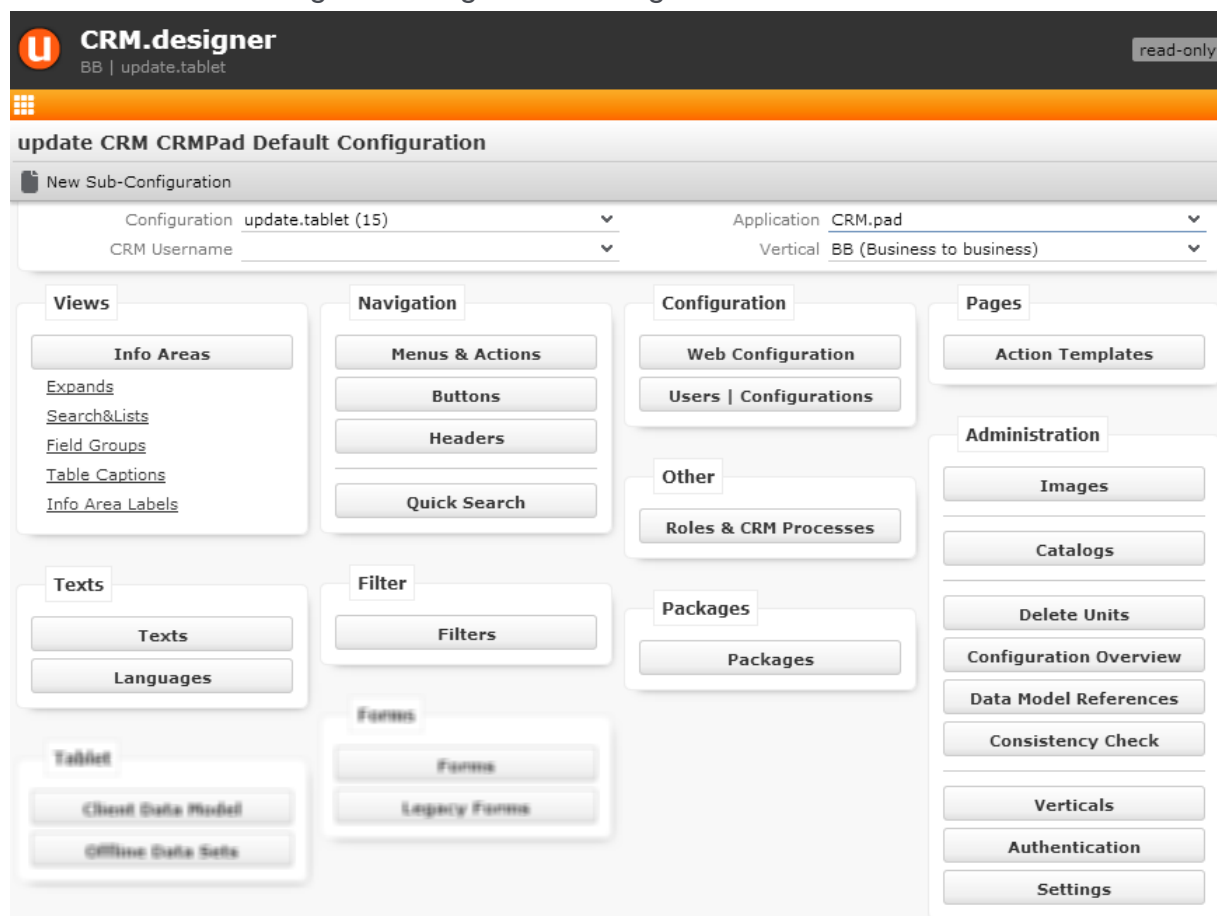
```
<update.server>
  <PlugIns>
    [...]
    <!-- add CRM.mobile support -->
    <PlugIn type="update.Mobile.Web.PlugIn,update.Mobile.Web" required="false"
  />
  </PlugIns>
```

On the client, CRM.mobile is accessed by adding `/mobile` to the URL used for update.CRM web, e.g. `http://mycompanyserver/updateCRM_web/mobile`

For a list of currently supported devices and operating systems. For further details, see [System requirements & supported devices](#) on page 8.

CRM.designer

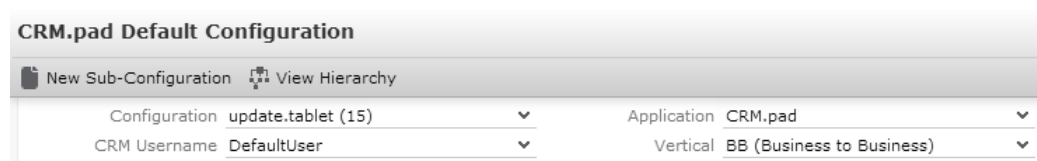
Learn how to configure using CRM.designer.



CRM.mobile shares its default configuration (update.tablet) with CRM.pad. However, not all items visible on the CRM.designer main page apply to both CRM.mobile and CRM.pad. Items available for CRM.pad only are blurred on the above screenshot.

Users configuration

Users and its parent configuration control the behavior and functionality of CRM.mobile.




The default configuration used by CRM.pad and CRM.mobile is called update.tablet. This is a root configuration which is not connected to UPDATE_DEFAULT used for update.CRM web.

You cannot change or delete the `update.tablet` configuration.

To create a custom configuration:

1. On the **CRM.designer** main page, select **CRM.pad** from the application drop-down list. The configuration field is automatically updated to `update.tablet`.
2. Click **New Sub-Configuration**.
3. On the **Parent Configurations** page, enter a **Configuration Name** and **Description**.
4. Click **Save**.

Since it is not possible for a user to change configuration units in CRM.pad and CRM.mobile (define lists, store queries, etc.), the automatic creation of a user configuration is not supported. All user configurations have to be created and configured by the administrator. If you want to use one custom configuration for all users, assign this configuration to the `DefaultUser`.



The screenshot shows the 'Users / Configurations' page with a breadcrumb trail 'BB | update.tablet | mobile_customized_parent'. Below the breadcrumb, there are tabs for 'Startpage', 'Users', and 'Configurations'. A 'Default Users' section contains a table with the following data:

	Default User Name	Configuration	On Change	Parent	Application
	DefaultUser	<u>customized_parent</u>	create new config	STATIC	update.CRM web
	DefaultUser	<u>mobile_customized_parent</u>	create new config	STATIC	CRM.pad
	DefaultUser	<u>update.ras</u>	create new config	STATIC	CRM.RAS

Below the table is a button labeled 'Users | Configurations'.

To create a user:

1. On the **CRM.designer** main page in the **Configuration** area, click **Users | Configurations**.
2. In the **User** area enter the user's login name as **User Name**.
3. Select a **Configuration**. The **Create New Config** flag is not supported by CRM.pad and CRM.mobile.
4. Select **CRM.pad** as **Application**.



The form shows a user creation interface with the following fields and options:

- A plus sign icon (+) on the left.
- A text input field containing 'JOHN'.
- A dropdown menu with 'mobile_customized_parent' selected.
- Two checkboxes: 'create new config' (unchecked) and 'dynamic' (unchecked).
- A dropdown menu with 'CRM.pad' selected.

5. Click  (Add).

Web configuration

Web Configuration parameters are key-value pairs that define the behavior of CRM.mobile.

To open **Web Configuration** page:


1. Goto **CRM.designer** main page in the **Configuration** area.
2. Click **Web Configuration**.

Web Configuration parameters defined in `UPDATE_DEFAULT` are not applied to CRM.mobile.

To use parameters from update.CRM web that are supported in CRM.mobile, you need to create and set them in update.tablet.

Web configuration parameters


CRM.mobile supports web configuration. The following Web Configuration parameters supported by CRM.mobile:

Note: Web Configuration parameters that are not part of update.tablet and need to be created are marked by an  icon.

Calendar parameters

Parameters	Description
CalendarFirstWorkingHour	Defines the start time of the work day. Default: 8
CalendarFreeTimeSlotMinimumHours	Defines the minimum time (in hours) required for a free time slot to be displayed. Default: 30 minutes
CalendarNumberOfWorkingHours	Defines the duration of the work day. Default: 10

Expand parameters

Parameters	Description
Expand.HideSubmitButtons	 <p>If checked, the additional Save and Cancel buttons are hidden from all New and Edit pages.</p>

History parameters

Parameters	Description
History.MaxEntries	<p>Defines the maximum number of entries in CRM.mobile's browsing history. For further details, see Tile-RecentlyUsed on page 63.</p> <p>If you filter the history e.g. for FI only (see infoAreaFilter), you may not see any records in the Recently Used tile as the history may contain lots of KP and MA records that are filtered by the action parameter.</p> <p>Default: 100 Type: Number</p>

Login parameters

Parameters	Description
Login.EnableLoginInDefaultLanguage	<p>By selecting this option the user can start the application in his default language, i.e. the language entered in his Rep record.</p> <p>If checked for the configuration assigned to the Default User, which is used for the login page, the option My Default Language (text group <code>Authentication</code>, text no. 0) is displayed in the language selection drop-downlist.</p> <p>This parameter is not part of update.tablet but needs to be created as Checkbox if required.</p> <p>Type: Checkbox</p>
Login.StoreCredentials	

Parameters	Description
	<p>If checked for the configuration assigned to the DefaultUser (which is used for the login page), the user's login name, password and login language are stored locally in the browser history. When accessing CRM.mobile the next time, the credentials are restored and the user is automatically logged on.</p> <p>The credentials are only deleted when clearing the browser history. The credentials are saved as plain text.</p> <p>Default: Unchecked</p> <p>Type: Checkbox</p>

Rep parameters

Parameters	Description
Rep.MaxResultsMobile	<p>Default number of reps displayed in the rep search, i.e. the drop-down list displayed for rep fields.</p> <p>Tapping on the "..." below the results loads the next <code>Rep.MaxResultsMobile</code> number of reps.</p> <p>Default: 20</p>

Search parameters

Parameters	Description
Search.MaxResultsMobile	<p>Default number of records displayed in lists.</p> <p>Tapping on the "..." below the results loads the next <code>Search.MaxResultsMobile</code> number of results.</p> <p>This setting is overridden by the respective setting in the action, e.g. <code>MaxResults</code> in <code>RecordListView</code> and <code>Tile:Search</code> actions.</p> <p>Default: 5</p> <p>Type: Number</p>

View parameters

Parameters	Description
View.HideEmptyFields	<p>If checked, then fields without values are hidden from the tabular Details view. For further details, see Tabular details view on page 32.</p> <p>Default: Unchecked.</p>
ShowTileFooterComplete	<p>If checked, tiles with long text that span multiple lines are displayed with a scrollbar at the footer.</p> <p>Type: Checkbox</p>

3

Using CRM.mobile

Learn to use CRM on Mobile and defining images for CRM Mobile.

For general information on using CRM Mobile, see [Administration](#) in the *Aurea CRM Web Administrator Guide*.

Images

Custom images that are supported for sprite images are allowed in CRM.mobile.

The images used by CRM.mobile are defined in the font (located at `..\styles\update.mobile.theme\fonts`).

In CRM.designer these images are currently not displayed. The `images.html` file in the `fonts` directory lists all available font images.

Adding custom images is only supported for sprite images. However, these images will not change their color according to the background which may lead to unwanted results.


Adding custom font images is currently not supported.

If no font images are displayed on a device (Windows Phone OS), you need to add the required mime type.

Info Area Customizations

Use the **Views** area of CRM.designer to customize the way records are displayed in Aurea CRM mobile.

To navigate to **Info Area**:

1. On the CRM.designer page, click **Info Areas**.
2. Search for the info area you want to customize and click  (edit). You will find all info area-specific configuration units listed.

You can design the following configuration units:

Field Groups

Field control types like details control, list control, and search control are used by CRM.mobile.

For general information on fields and field groups, see [Field Groups](#) in the *Aurea CRM Web Administrator Guide*.

The following sections lists all the settings supported by CRM.mobile:

Field Control Types

Learn about the field control types in Aurea CRM Mobile.

The following field control types are used by CRM.mobile:

Details Control

The Details control configures the contents of the Details Overview and the tabular Details view. In addition, it is used as a fallback if no Edit control is available.

As in update.CRM web, fields are grouped within panels, see [Configuring Panels](#) on page 22.

List Control

The List control configures which fields of a record are displayed in lists, and the sort order of the records.



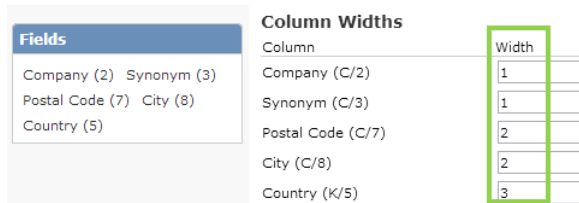
Each result row contains 3 lines (left-aligned). The order of the fields defined in the List control determines the order of fields in the result row.

You have the following possibilities to arrange the fields:

- Use the `Colspan` attribute to create combined fields.

Example in update.tablet: the company address in the `FI` List control. For further details, see [Field attributes](#) on page 25. This supports separators (blank, comma, colon, etc.) as well.

- Define **Columns Widths**:



Enter 1, 2 or 3 to specify the line where the value should be displayed. Several fields in one line are automatically separated by a blank.

If a field contains no value, it is hidden. i.e. no blank spaces are displayed. If the contents of a line exceed the available space, "..." is added (browser-dependent) and the remaining field values are omitted.

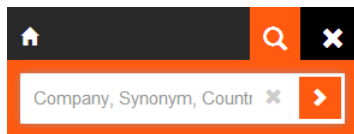
By default, fields are displayed without labels in a List control. To display a field label, **check** the `No label` attribute. For further details, see [Field attributes](#) on page 25.

In addition, you can use field functions in the List control to define field values as parameters to be used as default values.

Example in `update.tablet`: In the `MA` List control the **Starts on** field has **Function** = `Date`, **Starts at** has **Function** = `Time`. This allows you to use the start date and time of an activity as parameters (`$parDate:fdWeek+2d`, `$parTime:Hour`, etc.). For further details, see [Defining default values](#) on page 68.

Search Control

The Search control defines the search fields available in the info area search and for record selectors. for further details, see [Search&List Configuration](#) on page 35.



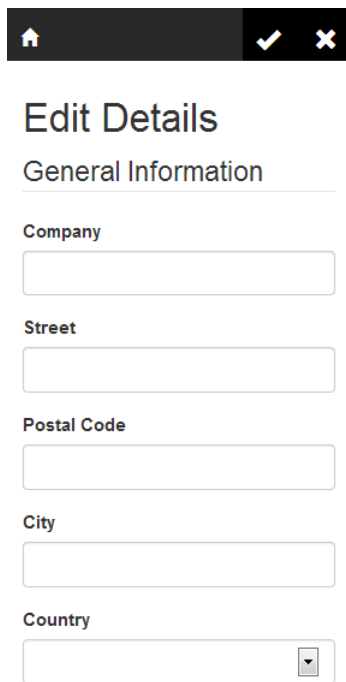
You can use fields of all types except

- decoded fields (Z-fields), e.g. a rep field,
- fixed catalog fields.

If you use such fields in a Search control, they will be ignored by the search (and hidden from the ghost text).

Edit Control

The Edit control configures the contents of the Details view when creating new records (Edit control in **New** mode) and when updating existing records (Edit control in **Update** mode).



If no Edit control is defined for an info area, the Details control is used as a fallback. You can define the following:

- The header text "Edit Details" uses text no. 1 from text group `Details_core`.
- You can group the fields on the control by defining group panels (📁), see [Configuring Panels](#) on page 22.

The panel's **Label** is displayed as sub-heading for the group, e.g. "General Information", "Company Details", etc.

- Add fields and set their attributes as in update.CRM web, see Customizing a Control in the CRM web Administrator Guide.

When using hierarchical catalogs, make sure to add both the parent and child catalog fields to the control. Users need to save the record before being able to select a value from a child catalog.

- Combined fields are displayed separately when in Edit mode.
- Z-fields without a `Select-Function` are automatically hidden when in Edit mode.

Special field groups

Special field groups are containers of user's input that you can configure in CRM.mobile.

CRM.mobile uses the following special field groups:

- `<infoAreaId>:Placeholders`: Contains placeholder definitions to be used for an info area
- `<infoAreaId>:RecentlyUsed`: The List control of this field group configures the list rows of the Recently Used tile (fallback: field group `<infoAreaId>`).

Field-Group 'FI:Placeholders', Control 'Details'

Save Delete

Search Fieldname or -number

0/S - FI-StaNo
1/L - FI-SerNo
2/C - **Company**
3/C - Synonym
4/C - Division
5/K - **Country**
6/C - County
7/C - **Postal Code**
8/C - **City**
9/C - Borough
10/C - **Street**
11/C - PO Box Postal Code
12/C - PO Box
13/Z - **Country Code**
14/C - **Area Code**
15/C - **Telephone**
16/C - Modem
17/C - Fax

'address' Street (10) Postal Code (7) City (8) Country (5)

'email' E-mail (18)

'phonenumber' Country Code (13) Area Code (14) Telephone (15)

'geo' Latitude (184) Longitude (185)

'mapdisplayname' Company (2)

Panel Editor

Label:

Name:

Type:

Add Fields to Panel:

CRM Processes:

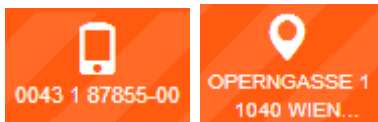
+

Placeholders

You can define a field group `<infoAreaId>:Placeholders` containing info area-specific placeholder definitions. Examples in `update.tablet: FI:Placeholders` and `KP:Placeholders`.

To use another than the default `<infoAreaId>:Placeholders` field group, specify your field group directly in the action parameter or label, e.g. `address = {FI_customPlaceholders/address}`

Note: The custom field group name must not contain a colon.



Placeholders are used:

- in labels of Quick Actions, e.g. `{phonenumber}` for the `MakeCall` button. For further details, see [MakeCall](#).
- as input arguments for buttons (defined as QuickActions and on tabular Details views), e.g. `address = {address}` in the `OpenMapAddress` button definition. For further details, see [OpenMapAddress](#).

Note: You can only use placeholders in actions that use a record ID (`uid`).

To define a placeholder:

1. Create a placeholder field group for the desired info area.
2. Create a Details control.

3. Create a new panel. **Name** is used as the placeholder's name (e.g. `address`, `phonenum` etc.).
4. Add the fields whose values should be used for the placeholder.

The value of the placeholder is created by combining all field values from the panel. You can use the syntax for combined fields to add spaces or delimiters.

If a placeholder evaluates to empty (e.g. the record contains no phone number), the button is hidden. You can define an alternate button to be displayed in this case. For further details, see [Alternate Buttons](#) on page 48.

Control Attributes

CRM.mobile currently supports the following control attribute:

Attributes

 tiles

Tiles:Company



Tiles

If defined for a Details control, a **Details Overview** page is displayed. For further details, see [Details overview](#) on page 34.

Specify the context menu containing the tiles, e.g. `Tiles:Company` in the Details control of field group FI, `Tiles:CompanyPerson` for KP, etc.

Configuring Panels

Fields displayed on details and edit controls are organized in group panels.

If you define a **Label** for a panel, it is displayed as a sub-header on the control.

The **Type** property allows to define special panels:

- `CHILDREN`: Configures sub-lists for a Details view. For further details, [Defining Sub-Lists](#).
- `PARTICIPANTS` / `LINKPARTICIPANTS`: Configures the participants area of activities (Details and Edit controls). For further details, see [Configuring the participants area for MA](#).
- `GRID`: CRM.pad uses `GRID`-panels to display fields in two columns. CRM.mobile displays the fields configured in a `GRID`-panel in one column on the Details view.

Defining Sub-Lists

The sub-lists on Details views (Related Data area) can be defined via additional panels in the Details control. This allows you to use the configurations you defined for CRM.pad.

To define a sub-list for a tabular Details view:

1. Add a new panel to the Details control.
2. Define a **Label** to be used as the header of the sub-list (fallbacks if undefined: header text from the Search&List configuration, info area name in plural).
3. Enter `CHILDREN` as **Type**.

Syntax:

```
CHILDREN_<S&Lconfig>_<maxRows>
```

or to use another than the default link (>> Link IDs in the update.CRM web Administrator Guide):

```
CHILDREN_<S&Lconfig>#<LinkID>_<maxRows>
```

If no Search&List configuration is found, a field group with the same name is used as a fallback.

If `<maxRows>` is omitted, the value of the `Search.MaxResultsMobile` Web Configuration parameter is used, >> [W](#).

Examples:

```
CHILDREN_MA_10
```

This defines a sub-list containing **Activity** records using the Search&List configuration `MA` and loading 10 records at a time.

```
CHILDREN_KM#2_10
```

This defines a sub-list containing **Ticket** records (using the Search&List configuration `KM`) where the current person is entered as **Contact Person**. Again 10 records are loaded at a time.


Configuring the participants area for MA

To define the participants area for an MA tabular Details view:

1. Add a new panel to the Details control.
2. Enter `PARTICIPANTS_<S&Lconfig>` as **Type**. In `update.tablet` the `MBPart` Search&List configuration is used.
3. Add the Participant ID field to the panel. This lists all external and internal participants using the specified Search&List configuration.

To define the participants area for an MA edit view (Edit control from `MA` and `MA:New`):

1. Add a new panel to the Edit control: Enter **Type** `PARTICIPANTS_<S&Lconfig>` as described above. This configures the internal participants area. The current user is automatically added as internal participant.

Tapping on  (**Add**) opens the Rep search which displays reps, groups and resources in this case. Groups are flagged with "(G)", resources with "(R)" in the result list. The texts no. 0 and 1 of text group `net_rep_types` are used.

2. Add panel of **Type** `LINKPARTICIPANTS_<S&Lconfig>`.

This configures the external participants area.

3. Add any text field to the panel.
4. Define a `Select-Function` for the field. For further details, see [Select Function](#).

In `update.tablet` the info area searches for FI and KP are used:

```
{"Type": "Record", "ContextMenu": "LIST_KP"}
{"Type": "Record", "ContextMenu": "LIST_FI"}
```

Users can add internal and external participants via the (hard-coded)  **Add** buttons. The **Participation Type** and **Status** icons are configured via field functions in the `MbPart` List control (`Requirement` and `Acceptance`). Tapping on an icon calls the menu action `Configuration:MAParticipants`.

Note: If you have defined (conditional) access rights for FI, KP or PE, make sure to also define them for the MB fields used in the External Participants panel. Otherwise a broken link (e.g. "REF:company") will be displayed instead of just hiding locked records.

Non-Field Elements

Understand the different field elements in CRM.mobile.

- Non-field elements (**LineFeed**, **Separator**, **Caption**, **Text**) are not supported by CRM.mobile. If used in a control, these elements are ignored.
- In CRM.mobile (and in CRM.pad) combined fields are defined via the `Colspan` attribute.
When in Edit mode, combined fields are displayed separately again.
- Linked fields are displayed read-only when in Edit mode.

For `NewView` actions linked fields are only supported if a record selector is defined as used in CRM.pad. For further details, see [Select-Function](#).

Example in `update.tablet`: The **Select Company** and **Select Person** fields in the `MA:New` Edit control.

Combined Fields

Learn how to define combined field in CRM Mobile.

In CRM.mobile and CRM.pad, combined fields are defined via the `Colspan` attribute. For further details, see [Colspan](#).

When in Edit mode, combined fields are displayed separately again.

Linked Fields

Learn to define the Linked field in Aurea CRM Web.

Linked fields are displayed read-only when in Edit mode.

For `NewView` actions linked fields are only supported if a record selector is defined as used in CRM.pad. For further details, see [Select-Function](#).

Example in update.tablet: The **Select Company** and **Select Person** fields in the `MA:New` Edit control.

Field attributes

Learn the supported field attributes of CRM.mobile.

The following field attributes are supported by CRM.mobile:

Bold / Italic: Displays the field's value in bold or italic. You cannot combine these attributes for one field.

Bold / Italic Label: Displays the field's label in bold or italic. You cannot combine these attributes for one field.

Don't save: The field content is not saved.

Hide Field: The field is hidden (no field label and field value are displayed).



Hyperlink: If checked, the field value is displayed as a hyperlink and the  icon is displayed beside the field (in the tabular Details view). Tapping the field value or the icon opens the website entered in the field in a new browser tab or window (browser-dependent).

Image: The `Image` attribute used to display images in CRM.pad is **not supported** by CRM.mobile.

Fields with the `Image` attribute set are **not displayed** in CRM.mobile.

No Label: If checked, the field's label is hidden.

Mail: If checked, the field value becomes a "MailTo:"-hyperlink and the  icon is displayed (in the tabular Details view). Tapping the field value or the icon opens the device's e-mail client and enters the field value in the **To** field.

If both the `Hyperlink` and `Mail` attributes are set for a field, the `Mail` attribute takes precedence.

Must-Field: If checked, the field label is displayed in red.

However, the field is not actually defined as mandatory by setting this attribute. You need to define mandatory fields in the data model (field property "must") or via the access rights definition, see [CRM.core Administrator Guide](#).

Readonly: If checked, the field is read-only and cannot be changed by the user. This field attribute is available for Details and Edit controls.

For fields defined as read-only in the access rights definition (**Fields** column > **Deny Update**) you do not need to set the `Readonly` field attribute.

Fields with **Deny New** in the access rights definition are displayed as read-only in Edit/New mode. In Edit/Update mode users may enter a value but the field is reset to empty on save.

Colspan: In CRM.pad and CRM.mobile the `Colspan` attribute is used to configure combined fields (in Details and List controls), e.g. define `Colspan = 2` to combine the current field with the next field. The fields are displayed in one line.

Example in update.tablet: the **Address** field in the `FI` Details control. To combine fields:

```
ADDRESS
Opengasse 17-21
1040 Wien
Austria
```

Field Color: Determines the color of the field's value in the tabular Details view.

Label Color: Determines the color of the field's label (tabular Details view, Edit view).

Multi Line: The field is displayed as a multi-line field.

If set for a Details control, the field displays the whole content (Text fields only).

If set for an Edit control, enter the number of lines as parameter (e.g. define `Multi Line = 5`). The field spans the defined number of lines (and becomes scrollable if the text is longer). Supported for Text and Number fields.

Select-Function: Use the `Select-Function` attribute (in Edit controls) to define record selectors. For more information, see [Record Selector Type Record](#) in the *CRM.pad Administrator Guide*.

You can define record selectors for reference fields (Z-fields) as well as for text fields (type: character).

Z-fields without a `Select-Function` defined are not displayed in CRM.mobile (Edit view).

For info area's linked to FI/KP, e.g. MA: Selecting a KP automatically enters the person's FI in the **Select Company** field.

If a company is selected first, the KP record selector automatically lists only that companies' KP records. To implement this you need to set `LinkRecord` (and `LinkId`) in your record selector definition (s. below).

Defining record selectors

Record selectors can be configured using action templates.

The following options are used to define record selectors:

- define a menu action using the `RecordListView` action template and setting the `Select-Function` attribute by directly entering the select function. For further details, see [JSON](#).
- define a menu action using the `RecordSelect` action template and setting the `Select-Function` attribute by just entering this menu action. For further details, see [RecordSelect](#). These record selectors can be reused in different info areas.

Configuration units

Defining a record selector requires configuration units to be configured.

The following configuration units are required for defining a record selector:

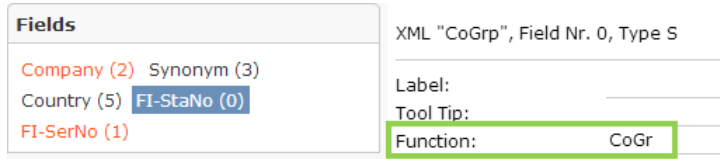
- a field group for the target info area (e.g. `FI:Select`) containing
 - a Search control configuring the search fields
 - a List control configuring the results list and field functions for copying/clearing field values.
- a Search&List configuration for the target info area (e.g. `FI:Select`) containing the above field group
- a context menu action using either
 - the `RecordListView` action template containing the ID (`InfoArea`) and Search&List configuration (`ConfigName`) of the target info area (e.g. `FISelect`), >> [JSON](#).
 - or
 - the `RecordSelect` action template containing the complete record selector definition (e.g. `F1Select`), >> [RecordSelect](#).
- in the Edit control of the current record's info area you can define copy fields as field functions, >> [Field functions](#).

Field functions

Record selectors use field functions for copying field values from the target record and clearing field values in the current record.

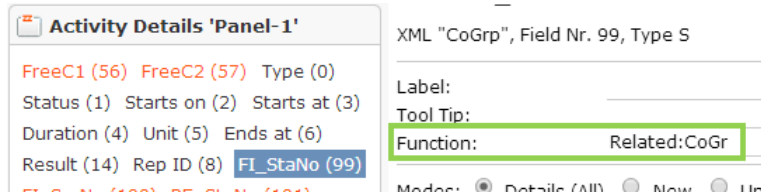
To configure this behavior:

1. In the List control of the target info area (e.g. `FI:Select`) define the fields whose values you want to copy:



The string you enter as **Function** is used as a parameter in the record selector definition. (You can enter any text; in this example, XML field names are used.)

2. In the Edit control of the current record's info area (e.g. `MA:New`), define the following functions to get the values from the target record:



The prefix ("Related:") is defined in the record selector definition (`TargetPrefix` parameter).

3. In the record selector definition or in the `RecordSelect` action, define which fields should be cleared if the user selects a new target record.

Note: For fields that are defined as link fields in the data model (type "reference (table)") you do not need to define the key fields (**StaNo**, **SerNo**) via field functions; these fields are automatically (re)set when selecting a target record.

JSON

You can use JSON syntax to define a record selector.

To define a record selector using the JSON syntax, go to edit control of the current record's info area (e.g. in `MA:New`) define the `Select-Function` attribute for the desired field.

Syntax

Parameter	Description
Type	The selector type: "Record"
ContextMenu	The context menu action (<code>RecordListView</code> action template) referencing the <code>Search&List</code> configuration of the target info area (<code>ConfigName</code> argument). For further details, see RecordListView on page 57. You can use the <code>FilterName</code> argument to apply a fixed filter to the search.
TargetPrefix	The prefix for the function names used to clear/copy fields. For further details, see Field functions . You can define any text.

Parameter	Description
Clear	An array of fields to be cleared if a new target record is selected. Enter <code>TargetPrefix</code> plus Function (as defined in the target info area's List control). For further details, see Field functions .
TargetLinkInfoAreaId	The ID of the target info area.
TargetLinkId	The link ID to be used for searching the target info area (if you want to use another than the default link -1). Note: If your record selector searches via the default link, you do not need to explicitly set <code>TargetLinkInfoAreaId</code> and <code>TargetLinkId</code> since the info area is determined by the <code>InfoArea</code> argument of <code>RecordListView</code> . However you can use these parameters to provide context information in the record selector definition.
LinkRecord	An optional parent of the target record. You must specify <code>LinkRecord</code> (and <code>LinkId</code> if another than the default link should be used) to display only target records from a specific parent (e.g. KPs for a selected FI).
LinkId	The link ID for the parent record. Example: Defining a record selector in KM to get only KP records linked to the company entered as Contact Company : " <code>TargetLinkInfoAreaId</code> ":"KP", " <code>TargetLinkId</code> ":"-1", " <code>LinkRecord</code> ":"FI", " <code>LinkId</code> ":" 2 "

Example

```
{ "Type": "Record",
  "ContextMenu": "KPfromFISelect",
  "TargetPrefix": "Related:",
  "Clear": ["Related:PeNo", "Related:PeGr", "Related:LastName"],
  "LinkRecord": "FI",
  "TargetLinkInfoAreaId": "KP",
  "TargetLinkId": "-1"
}
```

In `update.tablet` this record selector is defined for the **Select Person** field (KP:2) in the Edit control of field group `MA:New` (used by the `AddAppointmentfromList` button).

RecordSelect

You can define a record selector using `RecordSelect` action.

To define a record selector using the `RecordSelect` action:

1. Define a context menu action, e.g. `F1Select`.
2. Enter the name of the context menu action as `Select-Function`.

The `RecordSelect` action supports the following input arguments:

Parameter	Description
<code>ConfigName</code>	The name of the Search&List configuration of the target info area.
<code>FilterName</code>	The name of a fixed filter to apply to the search.
<code>TargetPrefix</code>	The prefix for the function names used to clear/copy fields, >> Field functions . You can define any text.
<code>ClearValues</code>	An array of fields to be cleared if a new target record is selected. Enter <code>TargetPrefix</code> plus Function (as defined in the target info area's List control), >> Field functions .
<code>TargetLinkInfoAreaId</code>	The ID of the target info area.
<code>TargetLinkId</code>	The link ID to be used for searching the target info area (if you want to use another than the default link <code>-1</code>), >> Link IDs in the update.CRM web Administrator Guide.
<code>LinkRecord</code>	An optional parent of the target record. You must specify <code>LinkRecord</code> (and <code>LinkId</code> if another than the default link should be used) to display only target records from a specific parent (e.g. KPs for a selected FI).
<code>LinkId</code>	The link ID for the parent record. Example: Defining a record selector in KM to get only KP records linked to the company entered as Contact Company : <pre>"TargetLinkInfoAreaId":"KP", "TargetLinkId":"-1", "LinkRecord":"FI", "LinkId":" 2 "</pre>

Example

Action for Menu F1Select

Action Template: RecordSelect

virtual page for RecordSelect configuration

Pass arguments as an object ?

Input Arguments

ConfigName	F1Quest
TargetLinkInfoAreaId	F1
ClearValues	"Related:QuestionnaireID"
TargetPrefix	Related:
FilterName	F1.Select

+ New Argumentname

In update.tablet this menu action is entered as Select-Function in the **Questionnaire** field in the Edit control of field group `MA:New`.



Expand configuration

The expand configuration of an info area contains the field group and header group that configure the Details view of a record.

In CRM.mobile two types of Details views are available:

- the Details Overview provides an overview of a record's contents using tiles. For further details, see [Details overview](#) on page 34.
- the tabular Details view lists field values of a record line by line. For further details, see [Tabular details view](#) on page 32.

Both views may contain Quick Actions, Extended Actions and sub-lists listing child records. For further details, see [Related data area](#) on page 35.

Users can switch between views by tapping the  /  buttons or the header text (Overview/Details).

The default view is the tabular Details view, i.e. if no Details Overview is configured for an info area, the tabular Details view is displayed and the arrow buttons are hidden.

Expand Page Configuration 'KP' (KP - Person in Company) BB (Business to business)

Startpage > All Info Areas > Info Area KP

Configuration for 'KP'

Field Group: KP Show KP

Header Group: KP Show KP

Table Caption: <not selected> -

Menu: <use default menu> Show Default

Image: Record:Person

Color: #ffb735

Info Area Default Action: SHOWRECORD

Description: Enter description here.

View Parent Relations 1. Contact Person (CP) Show all

LinkId:

Breadcrumb-Parent: Company (FI) Show all

LinkId:

Add Condition for this Field: Form of address (0)

Alternatives	Field	Cmp	Value	Alternative
+				Add an alternative

Save

The following options are available:

- **Field Group:** Defines which Details control is used.
- **Header Group:** Defines which `Expand` and `Expand.QuickActions` header is used.
- **Table Caption:** Defines the record's table caption.
- **Color:** The info area's color. Used for Record tiles, in lists and on Details views. If undefined, light gray is used.
- **Info Area Default Action:** Defines the default action that is executed when a user taps on a link (e.g. in a header). For further details, see [Default action](#) on page 39.
- **Breadcrumb-Parent:** Only the direct parent of the record is shown in CRM.mobile, no parents of the parent. Tapping on the link opens the Details view of the parent record using the info area's default field group `<infoAreaId>`.

Use **LinkId** to specify another than the default link.

The following chapters describe how these settings are used by the Details Overview and the tabular Details view.

Tabular details view

With tabular details view, you can see information in a table-like manner.

The tabular details view consists of the following items:

Header

- Breadcrumb parent: Defined by **Breadcrumb-Parent** in the Expand configuration. If undefined (and for records without a parent, e.g. FI), **Text** from the `Expand` header is used. For further details, see [Expand](#). If **Text** in the `Expand` header is undefined as well, no text is displayed.
- Buttons/Extended actions: Defined by the `Expand` header.
- Table caption: Defined by **Table Caption** in the Expand configuration. If undefined, the info area's default table caption `<infoAreaId>` is used.

Quick Actions

Defined by the `Expand.QuickActions` header from the **Header Group** specified in the Expand configuration.

Record Details

- The header text "Details" uses text no. 2 from text group `Details_core`.
- The Details control is determined by **Field Group** in the Expand configuration.
- The sub-headings ("General Information" etc.) and fields are configured by the panels (groups) and fields configured in the Details control.

You can use combined fields (`colspan`) and other field attributes. For further details, see [Field attributes](#) on page 25.

For long field values line breaks are automatically inserted.

For fields with special functions (phone, URL etc.) a button is displayed, allowing users to call the function. For further details, see [Fields with Actions](#) on page 33.



You can hide fields without values by checking the `View.HideEmptyFields` Web Configuration parameter.


Fields with Actions



Some fields provide special functions if configured in a tabular Details view. These functions are indicated by an icon displayed beside the field's value.

Tapping the value or the icon calls the function.

The following functions are supported:

- Links to a parent info area display the  icon. No additional configuration needed.
- Phone fields display the  icon, >> [MakeCall](#). You need to define a `Select-Function`:

```
{button:"MakeCall"}
```
- E-Mail fields display the  icon, >> [Email](#). No additional configuration needed (except placeholders s. note below).

- Website fields display the  icon. No additional configuration needed.
- For addresses you can configure the  icon, >> [OpenMapAddress](#). You need to define a `Select-Function`:

```
{button: "OpenMapAddress"}
```

Note: Notes: For fields that call a URL it depends on the used browser, whether the website or map is opened in a new browser window or in a new tab.

Note: For fields defined as Hyperlink or E-Mail fields in the data model, you do not need to set the respective field attributes.

Note: If a special field is empty (e.g. no telephone number entered), the icon is not displayed.

Note: For fields/icons using `MakeCall`, `Email` and `OpenMapAddress`, make sure the `<infoAreaId>:Placeholders` Details control contains all necessary placeholders, >> [Placeholders](#).

Details overview

The Details Overview (e.g. for FI in update.tablet) consists of the following items:

Header

Quick Actions

Defined as described for the tabular Details view. Record tiles display a field value. For further details, [Tile-Record](#) on page 64. Search tiles display the number of filtered records in a child info area. For further details, see [Tile-Search](#) on page 65.

Tapping on a Record tile opens the tabular Details view of the record, tapping on a Search tile lists the child records.

Filters for Search tiles are applied to the parent record.

- The header text "Overview" uses text no. 0 from text group `Details_core`.
- The info tiles are configured by a context menu, e.g. `Tiles:Company` for FI. Each context menu actions defines a tile. For further details, [Configuring a tile](#) on page 43 .
- The context menu must be referenced via a tiles attribute in the Details control. For further details, see [Control Attributes](#) on page 22.

Attributes

 tiles


Tiles:Company



Note: If no tiles are configured, the tabular Details view is displayed instead of the Details Overview.

Related data area

The related data area displays sub-lists listing child records of the current record.

Tapping on the sub-list header or on  opens a `RecordListView` of the child info area. Tapping on a list entry executes the default action, >> [Default action](#) on page 39.

The **Related Data** area (i.e. the) can be configured as follows:

- as panels of Type `CHILDREN` in the Details control determined by **Field Group** in the expand configuration. For further details, see [Defining Sub-Lists](#). This allows you to use the sub-list configurations you defined for CRM.pad.
- in the `Expand` header (**Infoareas** area). For further details, see [Expand](#). This allows you to use the tab configurations you defined for CRM.pad.

By default, the tabular Details view uses `CHILDREN`-panels, the Details Overview uses the definitions in the `Expand` header. If only one of the possible configuration methods is used, it is applied to both views (supported in version 8.1.7.2175 or higher).

Search&List Configuration

The Search&List configuration of an info area contains the field group and header group that configure the search fields and search results pages.

The Search&List configuration is used:

- by the global search.
- by the info area search.
- by Search tiles.
- for sub-lists.

Search&List Configuration 'FI' BB (Business to business)

Startpage > Search&List Configurations

Configuration for 'FI'

Infoarea **Company (FI)**
 Configuration my_mobile

Field Group [Show FI](#)

Header Group [Show FI](#)

Context Menu -

Default Action [Show SETTINGS SYNC](#)

Description

Help ID

Filter Name

The following options are available:

- **Field Group:** Defines the search fields (info area search only) and the contents of the search result rows. The Search and List controls of the specified field group are used.

If undefined, the info area's default field group `<infoAreaId>` is used.

- **Header Group:** Defines the header group of the `Search` header (info area search only). For further details, see [Search](#).

If undefined, the info area's default header group `<infoAreaId>` is used.

- **Default Action:** Defines the default action that is executed when a user taps on a row in a search results list. For further details, see [Default action](#) on page 39.
- **Filter Name:** Enter a filter name to apply a fixed filter to the search. For `RecordListView` and `Tile:Search` actions you can specify a fixed filter in the action definition as well. For further details, see [FilterName](#) and [Filters](#) on page 68.

If you specify filters in both locations, the filters are combined using logical AND.

Configuring the info area search

Learn how to configure the info area search in CRM.mobile.

The info area search can be called:

- from the application menu (`$Search`). For further details, see [Application Menu](#) on page 40.

Menu actions using the `RecordListView` action are listed as search options below the global search.

Selecting a search option takes the user to the appropriate info area search.

- by tapping the header of a search results list in the global search results
- by tapping a Search tile
- by tapping a sub-list header in a record's Details view

and tapping  (**Search**) again.

The info area search consists of the following items:

- a search field containing the configured search fields as ghost text. For further details, see [Search Control](#).

The fields are searched using logical OR.

- a list of predefined filters, if available (defined in the `Search` header. For further details, see [Search](#))
- a **Back** option allowing users to return to the global search
- extended actions on the search results page, if available (defined in the `Search` header).

Search fields, filters and extended actions are determined by the field group and header group specified in the Search&List configuration. For further details, see [Search&List Configuration](#) on page 35.

The Search&List configuration to be used is determined as follows:

- For info area searches for which a `RecordListView` action is defined, the Search&List configuration entered as `ConfigName` is used.
- For info area searches for which no action is defined (i.e. callable from the global search results only), the default Search&List configuration `<infoAreaId>` used.
- For linked searches (from sub-lists), the Search&List configuration specified in the sub-list definition is used.

The number of records displayed in the search results list is determined by the `Search.MaxResultsMobile` Web Configuration parameter (default: 5). For further details, see [Search.MaxResultsMobile](#). Defining `MaxResults` in a `RecordListView` (or `Tile:Search`) action overrides the Web Configuration parameter.

CRM.mobile-Specific Configurations

Learn about the configurations specific to CRM.mobile.

Since CRM.pad and CRM.mobile share the update.tablet configuration, some configurations may be optimized for being displayed on an iPad but not on a smaller mobile device.

You can define a CRM.mobile-specific Expand or Search&List configuration that should be used when viewing the application on a mobile device:

1. Create a configuration using the suffix `.small` (as copy of the existing configuration) and remove/hide all fields and functions that should not be visible on the mobile device.

If a configuration called `<config_name>.small` is found in addition to `<config_name>`, `<config_name>.small` is used instead of `<config_name>` on mobile devices.

Restriction: The functionality for alternative Expand configurations is not aware of the `.small` naming convention (>> Alternative Expand Configurations in the update.CRM web Administrator Guide). Therefore to use alternative Expand configurations, you need to create e.g. an `MA.small` configuration as a copy of `MA` and specify `MA.Visit.small`, `MA.Tel.small` etc. as alternatives in `MA.small`.

Table captions

Define table captions as in update.CRM web.

In CRM.mobile table captions are used:

- by `Expand` headers (as header text and breadcrumb parent). For further details, see [Expand](#)
- to define the text on Record tiles. For further details, see [Tile-Record](#) on page 64

You can add fields from the current info area and from linked parent info areas. You can define alternate table captions in case one or more fields you use do not contain a value.

Syntax

Use `{1}`, `{2}`, etc. to display the respective field values.

Use `{label:1}`, etc. to display a field's label.

For Boolean fields on Record tiles: Use `{bool:1}` to display the Boolean field's label if the field is set to true. If it is set to false, nothing is displayed. You can add an image (e.g. `Icon:Check`) to the table caption, that is displayed if the field is set to true.

You can add line breaks, blanks and separator characters (colon, comma, etc.).

Info Area Labels

Currently no plural info area names are defined in update.tablet. Since these names are used as a fallback e.g. for sub-list headers, you need to define them on the **Info Area Labels** page, if needed.

Navigation customizations

Configure navigations in CRM.mobile based on your user's needs.

The following user interface elements allow users to navigate in CRM.mobile:

- **Tabs:** If more than one page is defined for the start page, these pages are displayed as tabs. For further details, see [\\$Home](#).
- **Tiles:** On dashboard and details overview pages, tiles are used to display information (e.g. number of records or the value of a field) and can be used to access a record or list of records. They are configured as context menu actions. For further details, see [Configuring a tile](#) on page 43.
- **Buttons in headers or in tabular details views.** For further details, see [Buttons](#) on page 45 and [Fields with Actions](#) on page 33.
- **Quick Actions:** Buttons displayed in the **Quick Actions** bar, allowing users to quickly access context-specific functions (call phone number, add record etc.). For further details, see [Quick actions](#) on page 46.
- **Extended Actions:** Additional context-specific buttons. For further details, see [Extended actions](#).
- **Records in a list.**
- **Links to other records, external websites etc.**

For these elements you can define actions that are executed if the user taps the element.

Tabs, tiles and buttons can be displayed/hidden by assigning them to CRM Processes.

Alternatively, tiles can be hidden based on the current user's access rights, i.e. by setting the Info Area option in the menu action. Example: **Tile:NextAppointments** can only be accessed by users with access to the Activity info area (MA).

Default action

Understand how default action is determined in CRM.mobile.

Default actions is determined as follows:

- The default action specified in the `Search&List` configuration.
- The default action specified in the `Expand` configuration.
- The `SHOWRECORD` action.

Menus

The menus page allows you to customize the menus and menu actions used in CRM.mobile.

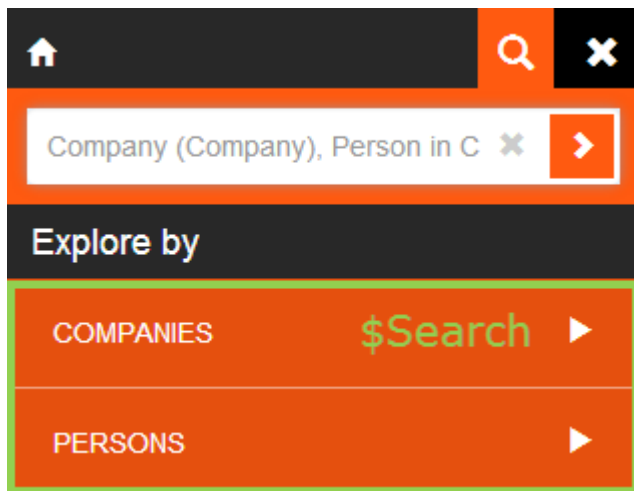
CRM.mobile uses menus and menu actions to configure the following:

- Application menu `SMARTBOOK`: The application menu configures the info area search. For further details, see [Application Menu](#) on page 40.
- Context menus are used to define the content of pages displayed in CRM.mobile, e.g. the start page or the tiles displayed on the tabs of the start page. For further details, see [Context menus](#) on page 42.
- Context menus are also used to define the info tiles on a record's Details Overview. Further details, [Details overview](#) on page 34.

Application Menu

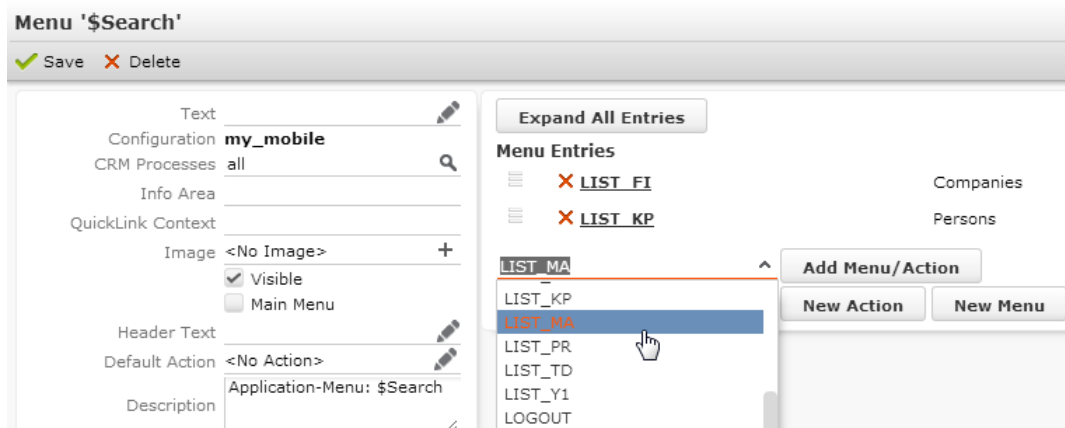
In CRM.mobile the application menu configures the info area search accessible from the start page.

The default application menu `SMARTBOOK` is used by both CRM.pad and CRM.mobile. For CRM.mobile only the `List_*` menu actions calling a `RecordListView` action are relevant. All other menu actions are disregarded by CRM.mobile.

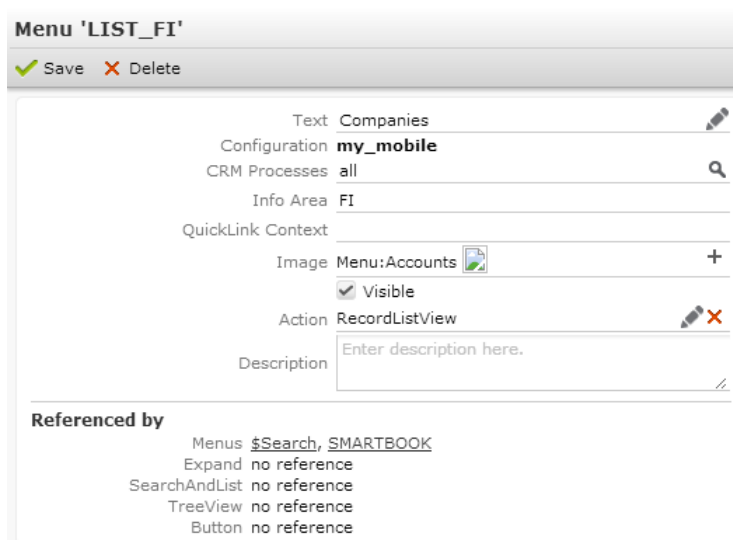


To use another than the default application menu for CRM.mobile, define a new application menu `$Search`. If `$Search` exists, CRM.mobile uses it instead of `SMARTBOOK`.

To configure info area search options to be displayed below the global search, you need to create or select menu entries for the desired info areas in the application menu (`SMARTBOOK` or `$Search`):



Only menu actions calling a `RecordListView` action are taken into account:



The following options are available:

- **Text:** The text for the search option (i.e. info area) in drop-down list, e.g. "Companies".
- **CRM Processes:** Assign one or more CRM processes to the menu action, >> Assigning CRM Processes in the update.CRM web Administrator Guide.
- **Info Area:** Use this option to hide a search option depending on the current user's access rights (**Deny Access**), e.g. the offer search is only available for users with access to info area PR.

Note: Restricting the visibility of a search option based on specific info area rights (e.g. via `MA:update,new`) is currently not supported by CRM.mobile.

- **Action:** Define the info area search using the `RecordListView` action. For further details, see [RecordListView](#) on page 57 and [Search&List Configuration](#) on page 35.

Context menus

Context menus are used to configure which tiles are displayed on a page.

\$Home

The context menu `$Home` defines the contents of CRM.mobile's start page.

The menu's **Text** is displayed as the **header text**: "CRM.mobile". Alternatively, you can specify a header text in the special header `Home.Header`. For further details, see [Home.Header](#). If texts are specified in both the menu and the header, the text from the menu is used.

The header **buttons** are defined by the special header `Home.Header`.

The **Quick Actions** available on the **Home** tab are defined by the special header `Home.QuickActions`. For further details, see [Home.QuickActions](#). If no Quick Actions have been defined, the area is hidden.

The menu actions defined for `$Home` determine the **tabs** displayed on the start page. The first menu action (tab) is active by default.


If only one menu action is defined, the tabs are hidden.

By default, each tab spans 50% of the screen. If more than two tabs (menu actions) are defined, the additional tabs are displayed in one (or more) additional row(s).

In `update.tablet` the start page contains two tabs:

- **Home** configured by menu action `A_Tiles:Home` and context menu `Tiles:Home`.
- **Dashboard** configured by `A_Tiles:Dashboard` and context menu `Tiles:Dashboard`.

Calendar

The `Calendar` context menu is called by the  (**Calendar**) button. It contains the available calendar views as tabs:

- Month
- Agenda

The first menu entry is used as default view, i.e. Month in `update.tablet`.

The header area of the calendar page is configured by the special headers `Calendar.Header` and `Calendar.Tabs`.

Tiles-home

The **Home** tab of the start page is defined by the following configuration units:

- Context menu action `A_Tiles:Home` using the `Tiles` action template.

The **Text** of the menu action defines the tab's name ("Home").

Note: If no **Text** is defined, the tab is not displayed.

- Context menu `Tiles:Home` defining the contents of the page. The menu to be used is determined by the `name` argument of the `Tiles` action. The context menu actions define which tiles are displayed.

- `Tile:NextAppointments.`
- `Tile:RecentlyUsed.`

Displaying the favorites list is currently not supported in CRM.mobile.

- Special header `Home.QuickActions` defining the available Quick Actions. For further details, see [Quick actions](#) on page 46. The header to be used is determined by the `actions` argument of the `Tiles` action.

Tiles-dashboard

The **Dashboard** tab of the start page is defined by the following configuration units:

- Context menu action `A_Tiles:Dashboard` using the `Tiles` action template. For further details, see [Tiles](#) on page 58.
- Context menu `Tiles:Dashboard` defining the contents of the page. The menu to be used is determined by the `name` argument of the `Tiles` action. The context menu actions define which tiles are displayed (`Tile:Today'sActivities`, `Tile:ThisWeek'sActivities` etc.) For further details, see [Configuring a tile](#) on page 43.

You can only use Search tiles on dashboards. For further details, see [Tile-Search](#) on page 65 .

The `filter` argument of the `Tile:Search` action is used to filter the search results, e.g. `MA.MyToday` for the **Today's Activities** tile.

Configuring a tile

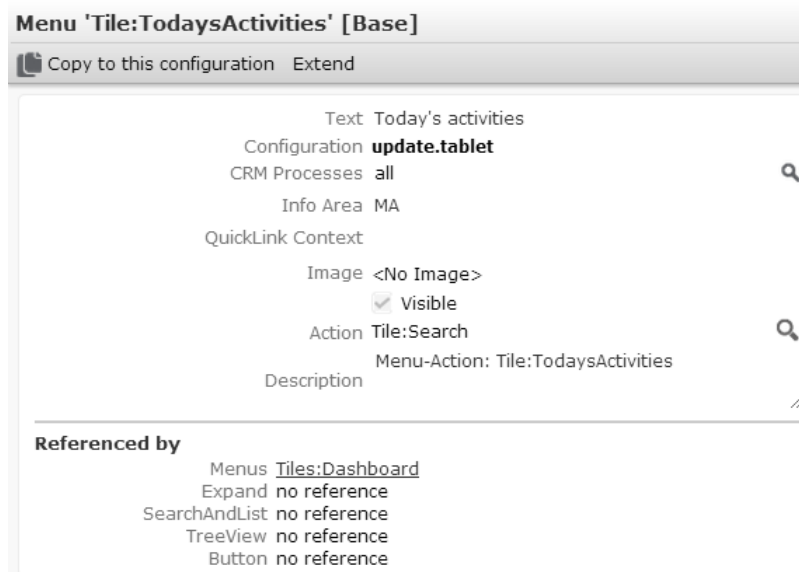
CRM.mobile lets you fully configure a tile like its visibility, text, image, and action.

Tiles are configured via menu actions using a `Tile:*` action template.

Make sure you follow the naming convention when creating your own tiles: `Tile:*` (e.g. `Tile:Today'sActivities`, `Tile:CompanyRevenue`, etc. in `update.tablet`).

To configure a tile:

1. Edit an existing menu action or create a new one.



2. Specify the following:

- **Text:** The label to be displayed on the tile. If undefined, the info area name is used (`{{infoarea:1}}`).

The following placeholders are available:

- `{1}` ...the result, i.e. usually the number of records found (for Search tiles). The number is displayed in the top left corner of the tile.
If no record is found, "0" is displayed, i.e. the tile is not hidden.
- `{infoarea:1}` ...the info area's name: Depending on the number of found records, the singular or plural is used.

The text is displayed in max. two lines, left- and bottom-aligned in uppercase. The font size is fixed, the line break is automatically inserted between words.

If the text length exceeds the available space, the text is clipped with "..."
(browser-dependent).

For Record tiles you define one or more fields to be displayed on the tile.

- **CRM Processes:** Assign one or more CRM processes to the menu action, >> Assigning CRM Processes in the update.CRM web Administrator Guide.
- **Info Area:** Use this option to hide a tile depending on the current user's access rights (**Deny Access**), e.g. `Tile:TodaysActivities` is only visible for users with access to info area MA.

Note: Notes: Make sure you set this option for all tiles that access info areas that you locked via the rights definition. Otherwise an error is displayed in CRM.mobile.

Note: Restricting the visibility of a tile based on specific info area rights (e.g. via `MA:update,new`) is currently not supported by CRM.mobile.

- **Image:** Select an image to be displayed on the tile. The images must be registered in CRM.designer. For further details, see [Images](#).

The image is displayed in the upper right corner of the tile.

For Record tiles you can alternatively specify images via action parameters. For further details, see [Tile-Record](#) on page 64.

- **Action:** Define the desired `Tile:*` action, e.g. use the `Tile:Search` action template for a Search tile or `Tile:Record` for a Record tile. For further details, see [Action templates](#) on page 55.

QuickLink Context and **Visible** are not supported by CRM.mobile.

3. Integrate the tile in the page where you want to display it. For further details, see [Arranging tiles on a page](#).

Arranging tiles on a page

The display of tiles on a page is determined by their order in the context menu: Depending on the defined size they are arranged in one or more rows. The tile size is determined by the `size` argument of the action (e.g. `Tile:Search`). For further details, see [size](#). The tiles are scaled dynamically according to the screen width.

Examples in `update.tablet: Tiles:Home, Tiles:Company`. You can add any number of tiles.

The list of tiles is dynamically reloaded when scrolling down.


Buttons

In CRM.mobile, you can configure buttons in headers.

Note: Make sure you only use actions that are supported by CRM.mobile. Otherwise the button will not be displayed. For further details, see [Action templates](#) on page 55.

CRM.mobile supports buttons in tabular details views. For further details, see [Fields with Actions](#) on page 33.



Each header may contain maximum of two buttons (right-aligned, the  **Home** button is hard-coded and therefore not counted here). If more than two header buttons are configured in a header, the **Menu** button is displayed instead beside the first button. Tapping on **Menu** displays the remaining buttons as Extended Actions. For further details, see [Extended actions](#).

Buttons without a label defined, are not displayed. A button without an image is automatically displayed in the menu, even if just this one button is configured in the header.

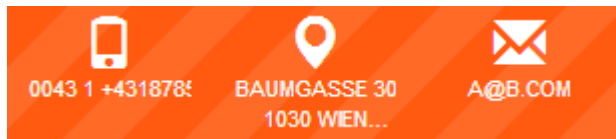
Grouping buttons (`GroupStart/End` and `GroupStartAdd`) are not taken into account by CRM.mobile, the buttons are automatically grouped as described above.

If more than two header buttons are configured in a header, the **Menu** button is displayed instead beside the first button. Tapping on **Menu** displays the remaining buttons as Extended Actions.

Make sure you only use buttons that make sense in the context, i.e. do not use a button that needs a record ID (e.g. `AddPerson`) in a company search results list.

Quick actions

Quick Actions allow users to quickly access context-specific functions like calling a phone number or adding a record.



Quick Actions are buttons configured in special headers `*.QuickActions`.

Quick Actions can be defined:

- for pages that use the `Tiles` action (in the `actions` parameter), e.g. `Home.QuickActions for A_Tiles:Home`.
- for Details and Details Overview pages by configuring a special header `Expand.QuickActions` in the header group referenced in the info area's `Expand` configuration, e.g. `Expand.QuickActions` from header group `FI` for the **Company** Details Overview.

You can configure max. three buttons at most. Additional buttons are ignored, i.e. not displayed.

If less than three buttons are defined, they are displayed left-aligned. If no Quick Actions are defined, the Quick Actions bar is hidden.


If a Quick Action requires a field value, e.g. the e-mail address of the record, and the current record does not contain that value, the Quick Action is hidden.


You can define any button to be used as a Quick Action. Make sure you only use buttons that make sense in the context, i.e. do not use a button that needs a record ID (e.g. `MakeCall`) on the **Home** tab.

Configuring a quick action

You can configure label, image, action, and CRM processes as quick actions.

Button 'AddAppointmentfromList' [Base]

 Copy to this configuration

Label	Add Activity	
Image	Button:Add 	
Action	NewView	🔍
Hot-Key	<input type="text"/>	
Query State Action		
CRM Processes	all	🔍
Help Text	<input type="text"/>	
Description	<input type="text" value="Enter description here."/>	//
<input type="checkbox"/> Hide button in QuickView <input type="checkbox"/> Hide button text <input type="checkbox"/> Owner Draw Button <input type="checkbox"/> Disable button <input type="checkbox"/> Hide button		
Referenced by		
Header	<u>Calendar.Header</u> , <u>Home.QuickActions</u> , <u>MA.Search</u>	

You can configure the following settings for buttons used as Quick Actions:

- **Label:** The label to be displayed on the button (max. 2 lines). You can specify any text and/or use placeholders. For further details, see [Placeholders](#). Long texts are truncated (browser-dependent).

If all placeholders specified in a button label evaluate to empty, the button is hidden, e.g. if a company record does not contain a phone number, the `MakeCall` button is not available for that record.

You can define an alternate button to be displayed instead of such an "empty" button by adding a `$AlternateButton` argument to the button's action definition. For further details, see [Alternate Buttons](#) on page 48.

If a label contains only text or if the specified placeholder is not defined in the `<infoAreaID>:Placeholders` field group used in the current context, the button is always enabled although it may do nothing.

- **Image:** Select an image to be displayed on the button. The images must be registered in CRM.designer. For further details, see [Images](#)
- **Action:** Define the action to be executed. For further details, see [Action templates](#) on page 55. You can provide values of input arguments by using placeholders. For further details, see [Placeholders](#).

For a list of buttons defined in `update.tablet`. For further details, see [Standard buttons](#) on page 48.

- **CRM Processes:** Assign one or more CRM processes to the button. For further details, see [Assigning CRM Processes](#) in the *Aurea.CRM Web Administrator Guide*.

Alternate Buttons

Alternate buttons are fallback for a button whenever they are disabled or hidden due to invalid values.




To create a fallback for a button that may be disabled/hidden due to invalid/empty placeholder values, add a `$AlternateButton` argument to the button's action definition. Enter the name of the alternate button as the value of `$AlternateButton`.

Example in `update.tablet`: The `OpenMapGeo` button has `OpenMapAddress` defined as alternate button, therefore if a record does not contain geo-coordinates, the record's address is used.


Standard buttons

This topic describes the standard buttons that you can create.

Following standard buttons can be created:

- The `Edit` button  is used to edit existing records. It uses the `OrganizerAction` action template with the `Action` argument set to `switchToEdit`. This calls an Edit page using the Edit control from the info area's default field group. All other input arguments are not supported by CRM.mobile.
- The `Save` button  is used to on Edit/New pages to submit changes and save the record. It uses the `Submit` action template. All input arguments are ignored by CRM.mobile.
- The `Cancel` button  is used to on Edit/New pages to undo all changes and return to the previous page. It uses the `Cancel` action template.
- The `Delete` button is used to delete the current record. It uses the `OrganizerAction` action template with the `Action` argument set to `deleteRecord`.

In `update.tablet` the `Delete` button is currently not used.

- The `Email` button  is used to open the device's email client. The current record's e-mail address is automatically entered in the **To** field. If configured as a Quick Action (in `update.tablet` e.g. in `Expand.QuickActions` header of header group `FI`), the `Email` button displays the current record's e-mail address in its label.

The `Email` button uses the `OpenUrl` action template. The e-mail address is determined by the `url` argument, e.g. `url = mailto:{email}`.

For information on the `mailto:` protocol. For further details, see <http://tools.ietf.org/html/rfc2368>. To pass e.g. the e-mails subject as well, define `url` as `mailto:{email}?subject={subject}` and define the fields to create the subject text as an additional placeholder `subject`.

Note: Please note, that non-field elements are not supported, i.e. you can only use field values to create the subject text.

When using placeholders, make sure you have defined them for all info areas where you want to use the button (e.g. field group `FI:Placeholders`). For further details, see [Placeholders](#).

If the field(s) contain no value(s), the `Email` button is hidden.

The automatic creation of an MA record when writing and sending the e-mail is currently not supported by CRM.mobile.

- The `MakeCall` button  is used to call the current record's phone number.

If configured as a Quick Action (in `update.tablet` e.g. in `Expand.QuickActions` header of header group `FI`), the `MakeCall` button displays the current record's phone number in its label.


The `MakeCall` button uses the `MakeCall` action template. The record whose phone number is called is determined by the `uid` argument. The phone number is determined by the `phoneNumber` argument, e.g. `phoneNumber = {phoneNumber}`.

When using a placeholder, make sure you have defined it for all info areas where you want to use the button.

If the field(s) contain no value(s), the `MakeCall` button is hidden.

The other input arguments available for the `MakeCall` action are currently not supported by CRM.mobile.

The automatic creation of an MA record when calling a phone number is currently not supported by CRM.mobile.

- The `OpenMapAddress` button  is used to open the device's map application, enter the current record's address in the map application's search field and display the address on the map.


If configured as a Quick Action (in `update.tablet` e.g. in `Expand.QuickActions` header of header group `FI`), the `OpenMapAddress` button displays the current record's address in its label.

The `OpenMapAddress` button uses the `OpenMap` action template. The address is determined by the `address` argument, e.g. `address = {address}`.

When using a placeholder, make sure you have defined it for all info areas where you want to use the button.

If the field(s) contain no value(s), the `OpenMapAddress` button is hidden.

Retrieving the address from a parent record (e.g. FI address for a KP) is currently not supported by CRM.mobile.

- The `OpenMapGeo` button  is used to open the device's map application, enter the current record's coordinates in the map application's search field and display the location on the map.

The `OpenMapGeo` button uses the `OpenMap` action template. Available input arguments: **geo**

The coordinates of the record, e.g. `geo = {geo}`.

displayName

The text to be displayed on the pin in the map, e.g. `displayName = {mapdisplay-name}`.

When using placeholders, make sure you have defined it for all info areas where you want to use the button. For further details, see [Placeholders](#).

If no geo-coordinates are entered for a record, the `OpenMapAddress` button is used instead (`$AlternateButton = OpenMapAddress`). For further details, see [Alternate Buttons](#) on page 48.

Retrieving the geo-coordinates from a parent record (e.g. FI coordinates for a KP) is currently not supported by CRM.mobile.

Headers

Learn to configure headers in CRM.mobile.

For general information on defining headers, see [Headers](#) in the *Aurea CRM Web Administrator Guide*.

The following topics describe the available settings and which standard and special headers are used by CRM.mobile.

Standard headers

You can configure standard headers in CRM.mobile like the Search and Expand headers.

The below standard headers are available:

Search


The `Search` header defines the following:

- **Text:** The header text is used as list header for the search results list and as header text of the page. If undefined, the plural of the info area name is used for the search results list; the page header is left empty in that case.

If a search is started by tapping on a sub-list header in a record's Details view, the page header shows a link to the parent record. The text of the link is created using the parent info area's default table caption.

- **Buttons:** Extended actions available on a search results page, e.g. `AddCompany` for FI. For further details, see [Extended actions](#).

Note: **Notes:** Make sure you use/define extended actions that make sense in the current context.

Note: The  (**Search**) button is hard-coded and therefore not visible in the Search header configuration.

- **Infoareas:** Specify predefined filters available for the info area search:
 1. Select the info area from the drop-down list and click on + (**Add**).
 2. Select the desired filter from the **Filter** drop-down list.

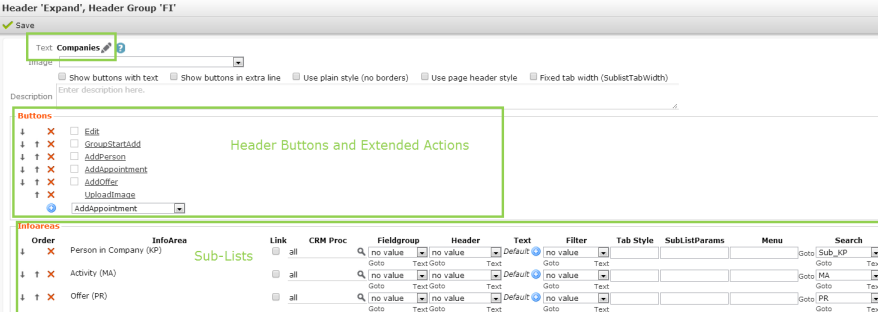
The name of the filter option displayed in the info area search is determined in the following order:

Text in the header definition.

Display-Text in the filter definition, the filter's name would be `FI.MYCustomers` in the example above.

Expand

The `Expand` header is used by Details Overviews and tabular Details views. For further details, see [Expand configuration](#) on page 31.



Order	InfoArea	Link	CRM Proc	Fieldgroup	Header	Text	Filter	Tab Style	SubListParams	Menu	Search
1	Person in Company (KP)	all		no value	no value	Default	no value			goto_Sub_KP	Text
1	Activity (MA)	all		no value	no value	Default	no value			goto_MA	Text
1	Offer (PR)	all		no value	no value	Default	no value			goto_PR	Text

The `Expand` header defines the following:

- **Text:** The header text is used as page header for the Details view. If undefined, no text is displayed.
For child records, Text is used, if no Breadcrumb Parent is defined in the Expand configuration.
- **Buttons:** Header buttons available for the Details view. For further details, see [Buttons](#) on page 45.
- **Infoareas:** Defines the Related Data area for Details views. For further details, see [Related data area](#) on page 35.

The following settings are available:

- **Link:** Check **Link** to select another than the default link. For further details, see [Link IDs](#) in the *Aurea.CRM Web Administrator Guide*.
- **CRM Proc:** Assign one or more CRM processes to the sub-info area. For further details, see [Assigning CRM Processes](#) in the *Aurea CRM Web Administrator Guide*.
- **Text:** Defines the header of the sub-list (fallbacks if undefined: header text from the Search&List configuration, info area name in plural).
- **Filter:** You can specify a filter to filter the child records.
- **Menu:** The menu action that is executed to display the sub-list, e.g. a `CalendarView` action for activity records (MA). For further details, see [CalendarView](#) on page 66.
- **Search:** The name of the Search&List configuration used by the sub-list. For further details, see [Search&List Configuration](#) on page 35. If undefined, the info area's default Search&List configuration `<infoAreaId>` is used.

Note: If an action is defined (**Menu** option), the Search&List configuration from that action is used, regardless of what is defined here.

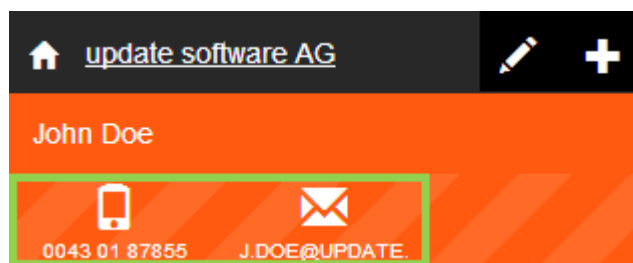
The other settings are not supported by CRM.mobile.

Special headers

Configure special headers in CRM.mobile like the Home.header, Home.QuickActions and Search.header.

Expand.QuickActions

This info area-specific header defines the Quick Actions available for that info area:



Home.header

This info area-independent header defines the topmost header in CRM.mobile:

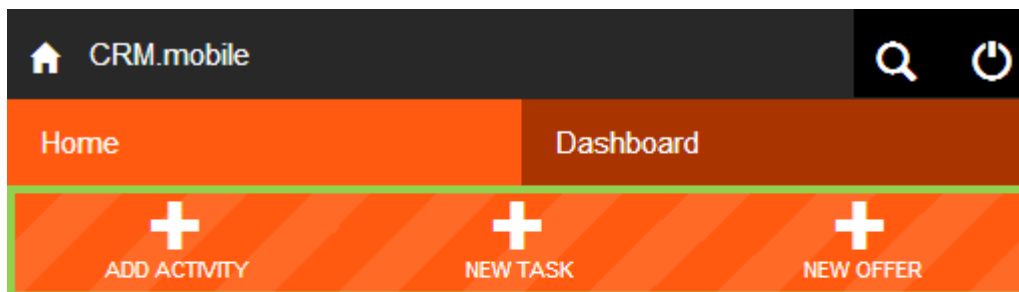


By default, the header contains the following items:

- **Home** button: not configurable (except for the image)
- Header text "CRM.mobile": configured by the **Text** defined for the context menu `$Home` (>> [\\$Home](#)) or by the header text. If texts are specified in both the menu and the header, the text from the menu is used.
- **Search** and **Logout** buttons: configured as header buttons in `Home.Header`.

Home.QuickActions

This info area-independent header defines the Quick Actions available on the **Home** tab:




MY NEXT ACTIVITIES



Search.Header

This info area-independent header defines the header of the global search results list. For details, see [Global Search](#) on page 53.

The  (**Search**) button is hard-coded and therefore not visible in the header configuration.

Make sure you only use buttons that do not require any context (record ID).

Global Search

The Global Search allows CRM.mobile users to search across multiple info areas by entering one search string.

It is called via the  (**Search**) button from the start page.

The ghost text indicates which fields are searched. The fields are searched using logical OR.

To configure the global search, click the **Quick Search** button in the **Navigation** area on the CRM.designer main page.

	Search Name	Fieldname	Tablename	Processes
✗ ↓	Company Search	Company	Company (FI)	all +
✗ ↑ ↓	Last Name Search	Last Name	Person in Company (KP)	all +
✗ ↑ ↓	First Name Search	First Name	Person in Company (KP)	all +
✗ ↑ ↓		Rep	Activity (MA)	all +
✗ ↑		Rep ID	Activity (MA)	all +
+	Add new Search			

To create a new global search entry:

1. Click **Add new Search**.
2. Under **Tablename** click + (**Add**).
3. Select the info area and field to be searched in.

As in update.CRM web, you can only use text, number and variable catalog fields in the search.

4. Click **Add Fields**.
5. On the Quick Search page click **Save**. (All other settings on this page are not used by CRM.mobile.)

In the ghost text the defined search field is displayed as "<info area> (<field name>)", e.g. "ItemMaster (Item name)".

The search results are listed per info area.

For the search results to be displayed the following configuration units must be available for all info areas that are part of the global search definition:

- a default Search&List configuration `<infoAreaId>`. For further details, see [Search&List Configuration](#) on page 35.
- this Search&List configuration must reference at least a field group containing a List control
- if the Search&List configuration references a header group as well, the header text of the `Search` header is used as list header for the search results.

Otherwise the plural of the info area name is used.

The page header is `Search.Header`.

The number of records displayed in each search results list is determined by the `Search.MaxResultsMobile` Web Configuration parameter. For further details, see [Web configuration](#) on page 13.

Default: 5

More results are loaded by tapping on the "... " below the list.

Action templates

Action templates are used to create different functions like calendar month, record list view and calendar view.

Action Templates

On the CRM.designer main page, click the Action Templates button to display all available action templates.

Action Template 'RecordListView' [Base]

Copy to this configuration

Callable from Application menu, context menu or process

Description

Help ID

URL/Function RecordListView

Options Pass arguments as object
 This action template is deprecated

Supported in CRM.pad
 CRM.mobile

Input Arguments

Name	EnabledFilter	Type	Value - Optional	Default
Description	Comma separated list of enabled filter names			

Name	ParentLink	Type	Value - Optional	Default
Description	The info area id of the link record (used together with argument "LinkRecord").			

The flags under supported in indicate where an action template is used (CRM.pad and/or CRM.mobile). Actions using an action template that is not supported by CRM.mobile, are not displayed in the application.

The following chapters provide information about CRM.mobile's standard action templates and their parameters.

Note: Action templates and parameters that are not listed here are not supported by CRM.mobile.


EditView-NewView

The EditView and NewView action templates are used to edit or create a record.

Examples in `update.tablet`: `AddAppointmentfromList` and `AddTDfromList` in `Home.QuickActions`, `AddCompany` in the `Search` header of header group `FI`, etc.

Edit and New pages are configured

- by the input arguments of the action. For further details, see [EditView-NewView](#) on page 56 and [NewView action parameters](#) on page 57.
- by the Edit control specified in the `ConfigName` argument or retrieved from the info area's default field group.

The **Edit** button  also calls an Edit page using the Edit control (or if undefined, the Details control) from the info area's default field group.

The page's header contains  (**Save**) and  (**Cancel**) buttons. At the bottom of the page, these buttons are available as well. You can globally hide these additional **Save** and **Cancel** buttons using the `Expand.HideSubmitButtons` Web Configuration parameter. For further details, see [Web configuration](#) on page 13.

The header text "Edit Details" uses text no. 1 from text group `Details_core`. Sub-headings are configured by the panels configured in the Edit control.

EditView-NewView action parameters

You can use parameters to fully configure EditView and NewView action..

`EditView` and `NewView` actions support the following input arguments:

SavedAction

Defines the action to be executed after the record is saved. If undefined, the edit-ed/newly created record is displayed in Details view after save.

Syntax:

`Menu:<MenuActionName>`

`Button:<ButtonName>`

Enter `Return` to return to the last page that was displayed before the Edit or New action.

Any other value is interpreted as the name of a menu action.

ConfigName

The name of the field group defining the layout of the page. The Edit control from this field group is used. For further details, see [Edit Control](#). If no Edit control is found, the Details control is used as a fallback.

If no Details control is found in this field group, or if `ConfigName` is undefined, the info area's default field group `<infoAreaId>` is used (again the Edit control or if undefined, the Details control).

If no valid control is found, an error message is displayed.

NewView action parameters

You can use parameters to fully configure NewView action.

In addition to the parameters listed here [EditView-NewView action parameters](#) on page 56, the `NewView` action (`mode = New`) supports the following input arguments:

LinkRecordId/LinkId/LinkRecordId2/LinkId2

`LinkRecordId`: Record to which the new record is linked (`Record` or `Linked`).

`LinkId`: ID of the link used for `link` (default=-1 i.e. the default link). This parameter is only relevant if multiple links between two info areas exist in the data model (e.g. multiple FI-links in KM). For further details, see [Link IDs](#) in the *Aurea CRM Web Administrator Guide*.

Use `LinkRecordId2` and `LinkId2` to specify a second link record.

TemplateFilterName

The name of a filter containing default values that are automatically entered in the new record. For further details, see [Defining default values](#) on page 68.

CopySourceFieldGroupName

The name of the field group used to determine variable default values from an existing record by using `$par` expressions. For further details, see [Defining default values](#) on page 68.

The List control of the field group is used, e.g. the `MA` List control in `update.tablet`.

CopyRecordId

The record (`Record` or `Linked`) from which the default values are read when using `CopySourceFieldGroupName`.

RecordListView

This topic has information about RecordListView action template.

The `RecordListView` action template is used to display a list of records.

For information on using `RecordListView` with record selectors. For further details, see [Select-Function](#).

RecordListView Action Parameters

The `RecordListView` action supports the following input arguments:

InfoArea

The ID of the info area to be searched.

This parameter is mandatory.

ConfigName

The name of the Search&List configuration to be used, see [Search&List Configuration](#) on page 35. If undefined or not found, the info area's default Search&List configuration `<infoAreaId>` is used.

LinkRecord/LinkID

`LinkRecord`: Optional parent record of the search results (for linked searches, e.g. all MA records linked to a KP): `Record` or `Linked`.

`LinkID`: ID of the link used for link (default=-1 i.e. the default link). This parameter is only relevant if multiple links between two info areas exist in the data model (e.g. multiple FI-links in KM). For further details, see [Link IDs](#) in the *Aurea CRM Web Administrator Guide*.

FilterName

Allows you to specify a fixed filter which is applied to the search and cannot be changed by the user. (The filter name is not visible for the user.)

If a Search&List configuration containing a filter is used, both filters are combined using logical AND. For further details, see [Search&List Configuration](#) on page 35.

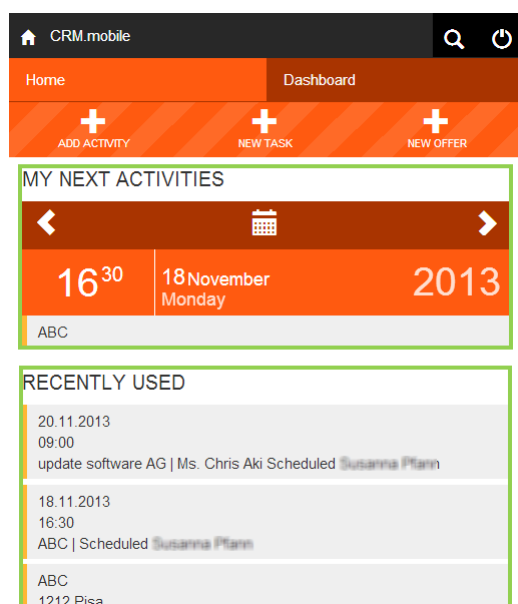
MaxResults

Determines the maximum number of results returned by the search. This parameter overrides the global value specified in the `Search.MaxResultsMobile` Web Configuration parameter. For further details, see [Search.MaxResultsMobile](#).

Tiles

The tiles action template is used to display tiles on a page.

Example in `update.tablet`: The menu action `A_Tiles:Home` (configuring the **Home** screen) uses the `Tiles` action template to display the tiles configured in the `Tiles:Home` context menu. For further details, see [Tiles-Home](#).



The `Tiles` action supports the following input arguments:

name

The name of the context menu that defines the tiles to be displayed, e.g. `Tiles:Home` for the **Home** screen, `Tiles:Dashboard` for the **Dashboard** screen.

This parameter is mandatory.


actions


The name of an optional header containing Quick Actions, e.g. `Home.QuickActions` for the **Home** tab. For further details, see [Quick actions](#) on page 46.



Tile-Calendar

This topic has information about Tile-Calendar action template.

The `Tile:Calendar` action template is used to display a Calendar tile, i.e. a mini calendar showing the next appointment of the current user.

Tapping on the  (**Calendar**) button displays the calendar, tapping on the record displays the Details Overview (or tabular Details view) of the record.

The  (**Calendar**) button is hard-coded and therefore not visible in CRM.designer. Its functionality is configured by the `Calendar` context menu. For further details, see [Calendar](#).

Users can navigate to past and future appointments using the   buttons.

In `update.tablet` the Calendar tile is used on the **Home** tab of the start page (context menu action `Tile:NextAppointments`).

Note: Currently, only MA records are taken into account.

If the user has no access rights for MA, the Calendar tile is hidden.

The `Tile:Calendar` action supports the following input arguments:

maxRows

The number of records (pre-)loaded at a time.

filter

By default, the MA records for the Calendar tile are filtered by Rep ID and Participant ID, i.e. only records where the current user is entered as rep and/or participant are taken into account.

Use the `filter` argument to apply a custom filter to the Calendar tile. This filter is then used **instead** of the default filter.

size

The size of the tile. If undefined, the default size is the full width of the screen.

Syntax

fieldGroupName

The name of the field group used for displaying the record, e.g. `MA.NextAppointments` in `update.tablet`. The List control of the field group is used.

If undefined, the default field group is used.

keepFirstStyle


If set to `true`, field formatting (via field attributes) is applied to the first field that is not empty, i.e. if you defined the first field of the List control e.g. as bold and the currently displayed record contains no value for that field, the bold style is applied to the first field that actually contains a value.

Tile-CalendarAgenda

This topic has information about Tile-CalendarAgenda action template.

The `Tile:CalendarAgenda` action template is used to display a Calendar Agenda tile, i.e. a list of upcoming appointments for the current user.

Currently, only MA records are taken into account.

Users can access the Agenda by tapping on the  (**Calendar**) button on the **Home** tab of the start page and selecting the **Agenda** tab (context menu action `A_Tiles:CalendarAgenda` in `update.tablet`). For further details, see [Calendar](#).

The entries are grouped by date. You can configure the date format to be used by creating and setting the `Locale.Date.DisplayFormat` and `Locale.Date.Separator` Web Configuration parameters in `update.tablet`. For further details, see [Locale Parameters](#) in the *Aurea CRM web Administrator Guide*. Currently, only short date formats are supported.

Each date heading contains a (hard-coded) **Add** button calling a `NewView` action that allows users to create new activity records. For further details, see [EditView-NewView](#) on page 56.

Tapping on a record in the list displays the Details Overview (or tabular Details view) of the record.

date

The agenda's start date. If undefined, the current date is used.

You can use placeholders to pass the date from the corresponding URL parameter to the action.

`{agendaDate}` uses the date currently selected on the Calendar tile (context menu action `Tile:NextAppointments`).

`{date}` uses the current date (since there is no way to pass another date to the calendar).

Example:

The date selected on the Calendar tile is passed to the URL as `/mobile/calendar?agendaDate=2014-12-10`

Specifying `date = {agendaDate}` in the `Tile:CalendarAgenda` action reads the date from the URL, i.e. the Calendar Agenda tile displays activities starting with the date the user selected on the Calendar tile.

readPastDays

If `date` is undefined, you can use `readPastDays` to specify the number of days prior to the current day that should be displayed in the agenda.

maxRows

Determines the maximum number of records displayed on the Calendar Agenda tile. This parameter overrides the global value specified in the `Search.MaxResultsMobile` Web Configuration parameter. For further details, see [Search.MaxResultsMobile](#).

size

The size of the tile. If undefined, the default size is the full width of the screen.

Syntax.

style

Applies a custom style (CSS class) to the tile.

filter

By default, the MA records for the Calendar Agenda tile are filtered by Rep ID and Participant ID, i.e. only records where the current user is entered as rep and/or participant are taken into account.

Use the `filter` argument to apply a custom filter to the Calendar Agenda tile. This filter is then used **instead** of the default filter.

fieldGroupName

The name of the field group used for displaying the records listed on the agenda. The List control of the field group is used.

If undefined, the default field group is used.

Tile-calendarMonth


This topic has information about Tile-calendarMonth action template.

The Tile:CalendarMonth action template is used to display a Calendar Month tile, i.e. a calendar month view with a list of appointments and free time slots for the current user on the selected day.

Currently, only MA records can be displayed.

The free time slots are calculated based on the `CalendarFirstWorkingHour`, `CalendarFreeTimeSlotMinimumHours` and `CalendarNumberOfWorkingHours` WebConfiguration parameters. For further details, see [Calendar Parameters](#).

The calendar month view is called by the context menu action `A_Tiles:CalendarMonth` in `update.tablet`. For further details, see [Calendar](#). Users can access the calendar

month view by tapping on the  (**Calendar**) button on the **Home** tab of the start page. The **Month** tab is opened by default.

The day that is selected in the calendar month view and each free time slot entry ("Idle from...") contain a (hard-coded) **Add** button (calling a `NewView` action) that allows users to create new activity records. For further details, see [EditView-NewView](#) on page 56.

Defaults for the start time of new activities:

- current day: current time
- all other days: start time of the first free time slot

Tapping on an activity record displays the Details Overview (or tabular Details view) of the record.

date

The date selected in the calendar month view (default: `{date}`). If undefined, the current date is used.

To define that the calendar month view opens to the date selected on the Calendar tile, specify `{agendaDate}` instead.

infoAreaId

The info area of the displayed records (default = MA).

Currently, only MA records are supported.

filter

Specify a filter for displaying MA records.

fieldGroupName

The name of the field group used for displaying the records listed in the calendar month view. The List control of the field group is used.

If undefined, the default field group is used.

Tile-RecentlyUsed

This topic has information about Tile-RecentlyUsed action template.

The Tile:RecentlyUsed action template is used to display a list of recently used records. A record is "used" as soon as it was displayed in Details view.

The `Tile:RecentlyUsed` action template is used to display a list of recently used records. A record is "used" as soon as it was displayed in Details view.

(Alternatively, you can use the `HistoryListView` action template which is supported in both CRM.pad and CRM.mobile.)

The result rows are configured via the List control of field group `<infoAreaId>:RecentlyUsed` (fallback: `<infoAreaId>`).

In update.tablet the Recently Used tile is used on the **Home** tab of the start page (context menu action `Tile:RecentlyUsed`).

The number of records saved in the browsing history is configured by the `History.MaxEntries` Web Configuration parameter.

As in update.CRM web, the browsing history is saved in the I9 info area. If no recently used records are available, the tile is hidden.

The `Tile:RecentlyUsed` action supports the following input arguments:

infoAreaFilter

Determines which records are displayed. If undefined, recently used records from all info areas are displayed.

Use the following syntax (JSON):

```
{ filter: ["FI", "KP", "MA"] }
```

size

The size of the tile. If undefined, the default size is the full width of the screen.

Syntax

pageSize

Determines the maximum number of records displayed on the Recently Used tile. This parameter overrides the global value specified in the `Search.MaxResultsMobile` Web Configuration parameter. For further details, see [Search.MaxResultsMobile](#).

For the `HistoryListView` action template, the `maxNumberHistoryRecords` is used instead.

Tile-Record

This topic has information about Tile-Record action template.

The Tile:Record action template is used to configure record tiles.

A Record tile can display one or more fields, field values and images. Tapping on the tile opens the record in tabular Details view.

Images on Record tiles are determined as follows:

1. dynamically based on a field value. For further details, see [imageMapFilter](#)
2. from the table caption. For further details, see [tableCaption](#)
3. from the image defined in the menu action.

The `Tile:Record` action supports the following input arguments:

uid

The unique ID of the record to be displayed (`Record` or `Linked`).

tableCaption



Defines the field(s) to be displayed on the tile. Enter the name of the table caption to be used.

You can specify one or more field(s) to be displayed (see [Syntax](#)), plus a fixed image. To display conditional images, use the `imageMapFilter` parameter. For further details, see [imageMapFilter](#).

If you add a line break (e.g. after the first field), the next fields/values are displayed in the next row using a smaller font.

If all fields from the table caption do not contain any values, an empty tile is displayed, i.e. the tile is not hidden.

imageMapFilter

Specifies a filter that contains the definition of conditional images, e.g. to show the different images for activity types.

The filter must contain the mapping of field values and images:

The image shows two parts of the application interface. The top part is a dialog box titled "Configuration for Filter 'MA.TypeImageMap'". It contains the following fields and options:

- Display Text: +
- New Date: 09.12.2013 12:39:23
- Upd Date: 09.12.2013 12:44:51
- Readonly:
- Invisible in lists (within the application):
- Include Associations: ?
- Description: Enter description here. (text input field)
- Save button

The bottom part of the image shows the filter configuration interface for "Activity (MA)". It displays a filter configuration with the following conditions:

- Filter: AND
- Condition 1: Type (0) = [dropdown]
 - Telefon CatCode = 0 (marked with a red X)
 - Besuch CatCode = 1 (marked with a red X)
 - E-Mail CatCode = 5 (marked with a red X)
- Condition 2: Image (Type (0)) = [dropdown]
 - Icon:Phone (marked with a red X)
 - Icon:User (marked with a red X)
 - Icon:Envelope (marked with a red X)

1. Define the values of the field in the first condition. (Only fields from the root info area are supported.)
2. Define the second condition for the same field with function Image.
3. Enter the names of the images.

Tile-Search

This topic has information about Tile-Search action template.

The Tile:Search action template is used to configure Search tiles on Details Overview pages and dashboards.

A Search tile displays the number of search results in a predefined search plus an explanatory text. Tapping on the tile displays a search page for the respective info area, listing the search results and allowing the user to change/refine the search.

The text and image displayed on the tile are configured by the menu action that is creating the tile. For further details, see [Configuring a tile](#) on page 43.

You can implement fixed filters for Search tiles:

- Enter the filter as **Filter Name** in the Search&List configuration.
- Enter the filter in the action (`filter` argument).

If you specified filters in both locations, the filters are combined using logical AND.

The applied filters are displayed in the header of the search page.

The `Tile:Search` action supports the following input arguments:

name

The name of the Search&List configuration to be used if a user taps the tile.

If undefined, the default Search&List configuration is used.

filter

A fixed filter used for the search result, e.g. `MA.MyToday` in `Tile:TodayActivities`.

If a Search&List configuration containing a filter is used, both filters are combined using logical AND. For further details see, [name](#) and [Search&List Configuration](#) on page 35.

link/linkID

`link`: Optional parent record of the search results (for linked searches, e.g. all MA records linked to a KP): `Record` or `Linked`.

`linkId`: ID of the link used for `link` (default=-1 i.e. the default link). This parameter is only relevant if multiple links between two info areas exist in the data model (e.g. multiple FI-links in KM). For further details, see [Link IDs](#) in the *Aurea CRM web Administrator Guide*.

style

The color of the tile. If undefined, the global default color `#eb5a19` is used.

size

The size of the tile. If undefined, the default size is the full width of the screen.

CalendarView

This topic has information about CalendarView action template.

The `CalendarView` action template can be used to display calendar items (e.g. activities) in sub-lists of the current company or person.

You need to define one action per info area (specified by the `InfoArea` action parameter, e.g. MA) and enter the action as **Menu** in the `Expand` header used by the parent record (e.g. FI or KP).

Definition of `FI Expand` header:

Header 'Expand', Header Group 'FI'

Save

Text **Companies**

Image

Show buttons with text
 Show buttons in extra line
 Use plain style (no borders)
 Use page header style
 Fixed tab width (SublistTabWidth)

Description
Enter description here.

Buttons

↓ × Edit
 ↓ ↑ × GroupStartAdd
 ↓ ↑ × AddPerson
 ↓ ↑ × AddAppointment
 ↑ × UploadImage
 AddAppointment

Infoareas

Order	InfoArea Link	CRM Proc	Fieldgroup	Header	Text	Filter	Tab Style	SubListParams	Menu	Search
↓ ×	Person in Company (KP)	all	no value	no value	Default	no value				Sub_KP
↓ ↑ ×	Activity (MA)	all	no value	no value	Default	no value			MA.CalendarView	no value
↑ ×	Offer (PR)	all	no value	no value	Default	no value				PR

Definition of MA.CalendarView action:

Menu 'MA.CalendarView' *

Save Delete

Text MA.CalendarView

Configuration **my_mobile**

CRM Processes all

Info Area

QuickLink Context

Action for Menu MA.CalendarView

Action Template CalendarView

Pass arguments as an object

Input Arguments

DefaultViewType	LIST
SortSequence	ASC
InfoArea	MA
ConfigName	MyMA
LinkRecord	Record
LinkId	
FilterName	<No Value>

Referenced by

Recent Changes

When

2014-09-30 16:02:21

In addition to the parameters described for the `RecordListView` action template, the `CalendarView` action supports the following input arguments:

DefaultViewType

Defines the default calendar view. Available values: `DAY`, `WEEK`, `MONTH` (default), `LIST`.

Regardless of the selected value, CRM.mobile always uses `LIST`, i.e. the calendar items are listed like in a `RecordListView` action.

SortSequence

The sort order used for the calendar items. Available values: `ASC` (default), `DESC`. The records are sorted by their date field, e.g. the **Starts on** field in MA (MA:2).

If the List control used by the action contains another sort field than the date field, sorting is applied as follows:

1. the sort order defined in the action (ASC or DESC by date)
2. the sort field from the List control, e.g. by Subject (MA:103).

FilterName

Allows you to specify a fixed filter which is applied to the CalendarView.

Alternatively you can define a filter called `<InfoAreaID>.CalendarView.Mobile`. This filter is applied to all `CalendarView` actions defined for the info area `<InfoAreaID>`.

`<InfoAreaID>.CalendarView.Mobile` overrules the filter defined in the action.

Filters

Learn about filters in CRM Mobile.

Filters are defined as in CRM web. `$cur` expressions are supported. For further information, see [Filters](#).

- `$curDate` is not supported; make sure you use `$curDay`.
- `$curTime` is supported and returns the same result as `$curTimeMin` (hh:mm). However, `$curTime` is currently not supported in CRM.win; make sure you do not use it in queries that are processed by CRM.server.
- Time calculations (e.g. `$curTimeMin+1h+30m`) are also currently not supported in CRM win.
- `$par` expressions are only supported by filters defining default values.

Defining default values

Filter definitions can be used to automatically enter default values when creating new records.

Note: Core default values defined in the CRM win Rights module are used when saving the record and only if the respective field is empty.

Configuration for Filter 'MA.TemplateNew'

Display-Text: +
 New-Date: 12.12.2013 17:16:46
 Readonly
 Invisible in lists (within the application)
 Include Associations
 Description

Activity (MA) + X

+ X Status (1) = [v] geplant CatCode = 0 X [input] ([input]) Para
 + X Type (0) = [v] Telefon CatCode = 0 X [input] ([input])
 Filter: + X AND [v] + X AND [v] + X AND [v] + X AND [v] + X Rep ID (8) = [v] \$curRep X [input] ([input])
 + X AND [v] + X Starts on (2) = [v] \$curDay X [input] ([input])
 + X AND [v] + X Starts at (3) = [v] \$curTime X [input] ([input])
 + X Create To-Do (133) = [v] TRUE X [input] ([input]) Parameter +

You can use catalog values, constants, \$cur expressions and \$par expressions (i.e. variables that are determined by a parent record).

For a complete description of the \$ expressions (syntax and examples), see [\\$cur expressions](#).

Note: If you want to use linked fields in a template filter, you need to pass the link to the field's info area in the action, otherwise, these fields will be ignored.

Boolean Fields

Any value except "N" evaluates to true.

Date/Time filters

In addition to \$cur expressions (\$curRep, \$curDay, etc.) you can use \$par expressions to get dates and times from parameters, e.g. \$parDate:fdWeek is the first day of the parameter's week. The parameters are defined in the info area's default List control/ see [Variable Defaults](#).

You can use date and time calculations for both \$cur and \$par expressions:

- \$curDay+1y+1d evaluates to tomorrow next year
- \$parDate:fdWeek+7d evaluates to first day of the week of Date plus 7 days.

Multiple Default Values

You can define multiple default values (fallbacks) for one field by adding \$onlyFirst as the first value:

Activity (MA) + X

Filter + X Starts on (2) = [v] \$onlyFirst X \$parDate X \$curDay X

The first expression that does not evaluate to null (or empty) is used. In the above example: If the parameter `Date` is defined and has a value (i.e. not empty), it is used. If `Date` is undefined or empty, the current date is used.

Fixed and variable defaults

You can use fixed default values or variable defaults to define the records you want to use.

Fixed Defaults

To use fixed default values (i.e. no `$par` expressions), you define the template filter and enter its name as `TemplateFilterName` in the `NewView` action. For further details, see [TemplateFilterName](#).

Example in `update.tablet`: `MA.TemplateNew` which is used for creating new activities (e.g. by the `AddAppointmentfromList` button).

Variable Defaults

To use variable defaults (i.e. defaults with values from an existing record), you need to configure the following:

1. Define the variables as parameters in a List control of the source info area. The List control's field group is then defined by `CopySourceFieldGroupName`. The parameters are defined as field functions.

Example: the `MA` List control in `update.tablet`

The screenshot shows two parts of the CRM mobile interface. On the left, a 'Fields' list contains several items: 'Starts on (2)', 'Starts at (3)', 'Subject (103)', 'Company (76)', 'Person (77)', 'Status (1)', 'Rep ID (8)', 'Type (0)', 'Ends at (6)', and 'Ends on (120)'. The 'Starts on (2)' item is highlighted with a blue background. On the right, a configuration window titled 'Field "Starts on"' is open. It shows the XML path 'XML "Date", Field Nr. 2, Type D'. Below this, there are input fields for 'Label:', 'Tool Tip:', and 'Function:'. The 'Function:' field is highlighted with a green border and contains the text 'Date'.

2. Define a filter using these variables in `$par` expressions, e.g. `$parDate:fdWeek`.
3. In the `NewView` action define the following input arguments:
 - a. Enter the name of your filter as `TemplateFilterName`.
 - b. Enter the field group containing your parameters (e.g. `MA` from the above example) as `CopySourceFieldGroupName`.
 - c. Set `CopyRecordId` to `Record` or `Linked`.

Languages and Texts

Learn about the Languages and Texts settings on CRM.mobile.

CRM.mobile uses the same language settings (Global Language Settings, Login Page Languages) as defined for CRM web, see [Languages](#) in the *CRM web Administrator Guide*.

Note: Not all languages listed on CRM.designer's Language Settings page are available for CRM.mobile.

CRM.mobile and CRM.pad use text groups and texts as defined in update.tablet. There is no connection to the text groups and texts defined in `UPDATE_DEFAULT`. However, the handling of texts is the same as in CRM web.