



Customer Relationship  
Management

# **CRM Client Administrator Guide**

**Version 2.6**

# 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

|                                                            |               |
|------------------------------------------------------------|---------------|
| <b>Preface.....</b>                                        | <b>5</b>      |
| About this documentation.....                              | 5             |
| Notation conventions.....                                  | 5             |
| Aurea global support.....                                  | 6             |
| <br><b>Chapter 1: Introduction.....</b>                    | <br><b>7</b>  |
| System Requirements.....                                   | 8             |
| Prerequisites.....                                         | 8             |
| Rights.....                                                | 9             |
| Restrictions.....                                          | 9             |
| Languages and Texts.....                                   | 9             |
| Installation and Setup.....                                | 10            |
| Mobile Services and IIS 7.0.....                           | 11            |
| Registering the Server.....                                | 11            |
| Authentication.....                                        | 15            |
| Performance Considerations.....                            | 15            |
| Monitoring CRM.Client Synchronization and Performance..... | 16            |
| <br><b>Chapter 2: Configuring CRM.Client.....</b>          | <br><b>18</b> |
| Users and Configurations.....                              | 18            |
| CRM.designer.....                                          | 19            |
| Data Model.....                                            | 20            |
| Offline Record Set (Offline Data Set).....                 | 22            |
| Offline Record Set (Offline Data Set) - Example.....       | 23            |
| Using JSON.....                                            | 27            |
| Login Page.....                                            | 28            |
| Adding an Insight Board to the Login Page.....             | 29            |
| Adding Buttons to the Start Page.....                      | 29            |
| Menus.....                                                 | 30            |
| Buttons.....                                               | 30            |
| Control Center.....                                        | 33            |
| Synchronization.....                                       | 33            |
| System Information.....                                    | 36            |
| Log Files.....                                             | 39            |
| Search.....                                                | 41            |
| Displaying Record Details.....                             | 45            |
| Displaying Data in Additional Tabs.....                    | 46            |
| Info Areas.....                                            | 47            |

---

---

|                                               |    |
|-----------------------------------------------|----|
| Field Groups.....                             | 47 |
| Search&List Configuration.....                | 57 |
| Headers.....                                  | 58 |
| Expand Configuration.....                     | 63 |
| Virtual Info Areas.....                       | 64 |
| Record Selectors.....                         | 65 |
| Filters.....                                  | 71 |
| Adding Filters to the Action Call.....        | 74 |
| Actions.....                                  | 75 |
| Action Templates.....                         | 76 |
| Web Configuration Parameters and Layouts..... | 89 |
| Client Timeout Parameters.....                | 90 |
| Format Parameters.....                        | 90 |
| Log Parameters.....                           | 90 |
| Login Parameters.....                         | 92 |
| Search Parameters.....                        | 92 |
| Sync Parameters.....                          | 93 |
| Other Parameters.....                         | 93 |
| Insight Board.....                            | 94 |

## **Chapter 3: Troubleshooting.....97**

|                                                         |     |
|---------------------------------------------------------|-----|
| General Checks.....                                     | 97  |
| Problems Concerning Rights.....                         | 97  |
| Connection Problems.....                                | 98  |
| Login Errors.....                                       | 98  |
| Performance Problems.....                               | 99  |
| Synchronization Problems.....                           | 100 |
| Synchronization Conflicts.....                          | 100 |
| Synchronization Timeout.....                            | 104 |
| Full Synchronization Terminates Due to the Timeout..... | 105 |
| Server Problems.....                                    | 110 |
| Lost Server Connection.....                             | 110 |

---

# 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               | <code>Fixed-width</code> font indicates code, path names, file names, environment variable names, parameter names, command names, machine names, URLs.                                                                                                   |
| <b>Bold Fixed-width</b>   | <b>Bold Fixed-width</b> font is used to indicate user input or to emphasize certain lines of code.                                                                                                                                                       |
| <i>Italic Fixed-width</i> | <i>Italic Fixed-width</i> font indicates a placeholder for which you must supply a value.                                                                                                                                                                |
| <b>Bold Sans serif</b>    | <b>Bold sans serif</b> typeface indicates the names of graphic user interface elements such as dialog boxes, buttons, and fields.                                                                                                                        |
| <i>Italic serif</i>       | In text, <i>italic 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 -LOGF. |
| [ ]                       | 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 <b>File &gt; &gt; Exit</b> means: "Open the <b>File</b> menu and choose <b>Exit</b> ." |
| >>            | 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

## Introduction

---

CRM.Client application allows you to access, display and edit data stored in Aurea CRM on your tablet PC, Desktop and mobile devices running on Windows 10 or iOS operating systems.

You can use CRM.Client along with Aurea CRM Web. The CRM.Client app works independently of the CRM Web version but both of them work with the same database. Your administrator has to define the configuration so that your CRM Web and CRM.Client work with the same database.

CRM.Client supports all available verticals and all industry solutions. It is configured with CRM.designer using the `update.tablet` configuration as the default configuration.

---

**Note:** CRM.Client shares its default configuration (`update.tablet`) with CRM.mobile. Therefore, keep in mind that changes to configuration units (e.g. a Details control or a placeholder definition) may affect the appearance and/or functionality of both applications.

---

---

: Apply the latest `update.tablet` configuration from CRM.Web version 11.2.22 or higher. The file contains the `smartbook:client_ui_text` which holds all the UI texts introduced by the user with CRM.Client. The fix ensures that the text is displayed in the language selected. Otherwise, all the interface reverts to the default texts in English.

---

Users can work with CRM.Client online as well as offline, therefore, data must be regularly synchronized.

---

**Important:** The CRM.Client Administrator guide provides details on the configuration nodes which are supported by CRM.Client. Feature and parameters that do not appear in this guide must be considered as not supported.

---

This document describes how to configure CRM.Client and covers topics such as:

- Server and client architecture
- Client user interface
- Data model and datasets
- Synchronization
- Setup

- Customization using CRM.designer
- CRM.Client specific configuration units

Additional information is provided in the following manuals:

- For details on installing Aurea CRM web and CRM.designer, >> Aurea CRM web Technical & Installation Guide.
- For general information on using CRM.designer and customizing web-based Aurea CRM applications, >> Aurea CRM web Administrator Guide.
- The administration and configuration of core functions are described in the CRM.core Administrator Guide.
- For information on the CRM.mobile configuration, >> CRM.mobile Administrator Guide.

All available documentation can be found on the Aurea CRM support page at <https://support.aurea.com>.

If not mentioned otherwise, all examples in this manual apply to the BTB vertical.

## System Requirements

System requirement for using Aurea CRM.Client.

---

**Note:** CRM.Client and CRM.Pad cannot be installed on the same device.

---

### Windows Devices

Windows Devices should be running Windows 10 OS (for Desktop) or Windows 10 Mobile OS (for mobiles and tablets). They are supported on ARM, x86, and x64 architectures.

### Apple iOS Devices

Apple devices running iOS 9 to iOS 11 are supported. The minimum device version supported is iPhone 5. iPad Mini and iPad Air 2 running a minimum of iOS 9 are supported.

## Prerequisites

Pre-requisites for using Aurea CRM.Client.

CRM.Client only works in combination with Aurea CRM web. An Aurea CRM web installation must exist as the Mobile Services server module uses the same process. Mobile Services server debug information can be found in the `web.log` file.

The following Aurea CRM web features also apply to CRM.Client:

- Users and configurations must already exist in Aurea CRM web. For information on how to create new users and sub-configurations, see *Aurea CRM web Administrator Guide*.
- Access rights are based on assigned roles, see *Aurea CRM web Administrator Guide*.

## Rights

Module and access rights for CRM.Client.

Check that the required module and access rights are assigned.

### Module Rights

The users need the following module rights assigned in order to access CRM.Client:

- Mobile Services (`mmri.exe` or `mmcfg.exe`)

### Access Rights

Rights defined in the Aurea CRM win Rights module also apply to Mobile Services:

- View rights (unconditional & conditional) are applied when the application synchronizes data.
- Edit rights (New, Update, Delete) are applied on the server when the user executes an operation.

The user is informed about rights violations when online and about synchronization conflicts, see [Synchronization Conflicts](#) on page 100.

## Restrictions

Learn about the restrictions in using CRM.Client.

CRM.Client on iOS devices does not support all of Safari's browser communication features.

For example, you cannot use `https` in combination with a certificate based on a root certificate that is **not** from a trusted authority (like Verisign).

## Languages and Texts

All available languages are included in the `update.tablet` configuration.

Some texts are displayed before a server connection can be established. If you use another language than English, German or French, these texts are displayed in English.

When users log on, they can select the desired language.

Languages and texts are handled by CRM.designer. Languages are installed with language patches, as for every other update application.

## Data Stored on the Device

The following data is stored on the device:

- A database containing CRM data (records and catalogs) in the user's language
- A database containing the application's configuration in the user's language
- A database containing the uncommitted data (records saved by the user but not yet synchronized with the server)
- Resource files transferred with the configuration
- Documents from the CRM database

You need to define the offline data model and offline record and data sets, see [Data Model](#) on page 20 and [Offline Record Set \(Offline Data Set\)](#) on page 22.

## Installation and Setup

Learn about CRM client installation and setup.

To be able to use CRM.Client, you need:

- The CRM.Client app for the user, available in the Apple iStore or the Windows iStore free of charge.
- Mobile Services (the name of the server application enabling the use of CRM.Client), included in the installation package of Aurea CRM web and in Aurea CRM web patches.

When an Aurea CRM web version is installed, Mobile Services is automatically installed as well, if it is included in your license. The `update.Mobile.Services.dll` is installed in the `\bin` directory and the `settings.xml` and `web.config` files are changed accordingly, see [Mobile Services and IIS 7.0](#) on page 11.

You need to enable the CGI restrictions for .net Framework (.net Framework is included in your Aurea CRM web installation). Start Aurea CRM web with .net Framework 4.0 and enable the CGI restrictions.

## Mobile Services and IIS 7.0

If Mobile services are installed on an IIS 7.0 server, HTTPHandlers are ignored and only handlers are used.

Therefore, you need not change the `httpHandlers` section of the `web.config` file, but must add the following entry underneath the `</system.web>` (closed system web) tag:

```
<system.webServer>
  <handlers>
    <add name="mobileServices" verb="POST,GET" path="mobile.axd"
type="update.web.Services.mobileServices,update.web" />
  </handlers>
</system.webServer>
```

## Registering the Server

To use CRM.Client you need to register your Aurea CRM web server.

The server can be registered either by entering the following URL into a web browser, or - if the URL was sent to you by e-mail - by clicking on the link:

```
crmclient://configureserver?identification=<IDENT>&name=<NAME>&url=<URL>
&authenticationType=username&useragent=crmclient
```

where,

- `crmclient://` is the URL protocol supported by the CRM.Client application.

---

**Note:** `crmpad://` URL protocol for CRM.Pad is also supported by CRM.Client.

---

- `<IDENT>` and `<NAME>` can be the same. `<IDENT>` is CRM.Client's internal ID of the server and `<NAME>` contains the name displayed in the login mask.
- `<URL>` contains the server's base URL, usually identical with the Aurea CRM web installation.
- `<AuthenticationType>` defines how the Mobile Services server connects to the Aurea CRM web server (`username` or `revolution`), see [Authentication](#) on page 15.
- You can also specify the user agent when registering your server. Add the `userAgent` parameter to the URL, for example, `userAgent=crmclient`.

For example:

```
crmclient://configureserver/?identification=LS-AureaCRM&name=LSAureaCRM&
authenticationType=username&url=http://acrm.eastus.cloudapp.azure.com/demo.lsi/
```

## Defining a Default Registration

Learn how to configure default registration.

Configuring a default server can be done locally on the device by setting the desired values in the `info.plist` file (located at `Products/Applications/CRM-client.app/` directory for iOS devices) or the `info.ini` (located at `C:\Program Files\WindowsApps\178E3AFA.AureaCRMClient_*****\Config\` directory for Windows devices).

The following parameters need to be configured depending on the default registration types:

- **com.update.CRMpad.defaultServer**

**Type:** String

**DefaultValue:** `CRM Revolution(crmpad://configureserver?identification=update&name=update& url=https://my.update.com/login&networkUsername=&networkPassword= &defaultErrorReportingEmailAddress=&userAgent=&authenticationType=revolution)`

- **com.update.CRMpad.defaultServer.registerFirst**

**Type:** Boolean

**Default Value:** False

- **com.update.CRMpad.defaultServer.force**

**Type:** Boolean

**Default Value:** False

The default registration types and the required parameter values are described below:

1. User tries to login without any server listed on the login screen.
  - If no server is registered yet as described in [Registering the Server](#) on page 11 then the server <https://my.update.com/> is registered automatically with "update" as <NAME>. This is applied in SaaS environments where you do not need to register a server.
  - User can set the configuration parameters to the following in the appropriate configuraton file:
    - com.update.CRMpad.defaultServer= appropriate value as described in [Registering the Server](#) on page 11.
    - com.update.CRMpad.defaultServer.registerFirst=False
    - com.update.CRMpad.defaultServer.force=False
2. Add the default server automatically to the list of servers the user is allowed to login.
  - com.update.CRMpad.defaultServer= appropriate value as described in [Registering the Server](#) on page 11.
  - com.update.CRMpad.defaultServer.registerFirst=True
  - com.update.CRMpad.defaultServer.force=False
3. Force the user to be allowed to login only to the default server.
  - com.update.CRMpad.defaultServer= appropriate value as described in [Registering the Server](#) on page 11.
  - com.update.CRMpad.defaultServer.registerFirst=False
  - com.update.CRMpad.defaultServer.force=True

## Including User Credentials in the Registration

Learn how to add user credentials in the registration.

You can also add the user's credentials to the registration URL. For example, if you want your users to access CRM.Client directly from another app.

If you allow access to CRM.Client directly, the user credentials must always be provided when the user logs in for the first time. The credentials are stored on the device.

The following options are available:

- `username`: the user's login name. Must be given if the `loginMode` option is used.
- `loginMode`: Defines how the login is carried out.
  - `savepasswordlogin`: The password is stored locally and the user is logged in automatically.
  - `savepassword`: The password is stored locally, but the login screen is displayed. The credentials are pre-filled, but the user must login manually

For example:

```
crmclient://configureserver?identification=CRMClient
&name=Sally&url=http://myserver/updatecrm &authenticationType=username
&username=Seller2&loginMode=savepasswordlogin
```

---

**Note:** Ensure that the `Login.SavePasswordAllowed` Web Configuration parameter is set, see [Login.SavePasswordAllowed](#).

---

## Displaying the Server Before the First Login

By default, servers to which you can log on to are only available on the login page after your first login.

Add the following to your registration link to register the default server automatically:

```
com.update.CRMclient.defaultServer.registerFirst=yes
```

## Removing the Server

Learn how to remove the server.

You can remove a registered server by entering the following URL into the browser:

```
crmclient://removeServer?name=<NAME>
```

Replace `<NAME>` with the server name that you have entered in the `name=<NAME>` parameter when you registered the server.

---

**Note:** `removeServer` does **not** remove the local database on the client. If you re-register the server using the same `<IDENT>` and `<NAME>` as before, CRM.Client connects to this existing database.

---

## Authentication

After starting CRM.Client, the user must enter the Aurea CRM web server (URL) to connect to, the user name and the corresponding password.

Depending on the server type, the following access types are available:

- **Anonymous Server Access:** Authentication to the CRM application is handled with the Aurea CRM credentials (user name and password, authentication type `username`).
- **Revolution Server Access:** Authenticate with e-mail address and password (authentication type `revolution`).

If login failed, you can check whether authentication was successful by entering one of the following URLs into Safari:

## Performance Considerations

The performance of synchronizing and searching data on the client depends on the size and complexity of the data model and the offline dataset.

If large data volumes are involved, synchronizing can be very time-consuming.

Each dataset is synchronized synchronously in one server request and must be transferred before a timeout occurs:

- The default client timeout is set to 60 seconds and can be changed, see [ClientRequestTimeout](#).
- The default ASP.NET server timeout is set to 90 seconds. You can change it with the Aurea CRM web installation. For further information refer to the *Aurea CRM web Installation Guide*.

---

**Note:** Bear in mind that setting these values higher might affect the application's performance.

---

For a thin connection (for example, GSM) the timeout hardly ever occurs. A timeout only occurs if time runs out before the first response package is transferred. A full synchronization on a thin connection takes a long time.

Do not use link fields ("Z fields") in your offline data set. Each offline dataset package is requested with one SQL statement, but the server might execute additional read operations for each record containing link fields.

When a dataset is requested, the following is applied and can result in complex SQL statements:

- The defined filters of each info area
- The current user's rights
- Dependencies to other info areas including defined filters and rights

---

**Note:** Complex SQL statements might need a long run time resulting in a timeout on the database server. This could result in incomplete record sets being returned to the client. For more information, see [Troubleshooting](#) on page 97.

---

---

**Note:** You can set the `RequestMode` option to `BEST` when assigning the page call for a Details control, then the record and its link are loaded from the server if a server connection is available.

---

To limit the duration of synchronization, you can:

- Define more than one offline data set for one info area, each containing only one filter or one dependency (the full timeout limit applies to each dataset, a record can be contained in more than one dataset).
- Define filters in a less complex way, even if this results in more records being transferred.
- Consider reducing your offline data set.
- Consider reducing the complexity of your rights definition, for example, by replacing them with triggered information (easier to query).

## Monitoring CRM.Client Synchronization and Performance

Learn how to monitor CRM.Client synchronization and performance.

You can capture the following detailed information for synchronization tasks performed by each user:

- Synchronization type
- Time taken for synchronization
- Total number of synchronizations performed by the user including the average time, max time etc
- Information per infoarea: Last synchronization date, start and end time, total time, and number of records synchronized.

To capture the log information you have to use the `update.Mobile.Services.Performance` filter in the `settings.xml` file for CRM.Web. Specify the path to the log file using the `filePath` attribute of the `<Writer>` element. Configure the attributes in the `<Channel>`, `<Writer>` and `<Formatter>` elements as required.

See the example configuration below for details:

```
<Channel facilityFilter="update.Mobile.Services.Performance"
logLevel="Debug"
logLevelDebug="Debug"
writerName="fileLogWriter4"
formatterName="webLogFormatter4"/>

<Writer filePath="log\performance.log"
name="fileLogWriter4"
maxSizeKB="8192"
closeOnIdleTimeout="5"
type="update.Lib.Logging.Writers.FileLogWriter,update.Lib" />

<Formatter name="webLogFormatter4"
logDate="true"
logTicks="false"
logFacility="false"
logLevelFormat="None"
logException="false"
logMemory="false"
type="update.Lib.Logging.Formatters.WebLogFormatter,update.Lib" />
```

The following screenshot shows the synchronization log information captured for a user:

|                                                                             |                         |                         |                         |                         |                         |                         |                         |
|-----------------------------------------------------------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|
| 2017-07-27;13:18:39; 868;SID[d822657a-2128-4c1d-9e0d-6583e0a1b625];TID[21]; |                         |                         |                         |                         |                         |                         |                         |
| User: SU                                                                    |                         |                         |                         |                         |                         |                         |                         |
| Sync Type: FullSync                                                         |                         |                         |                         |                         |                         |                         |                         |
| From: 2017-07-27 13:15:09,589                                               |                         |                         |                         |                         |                         |                         |                         |
| To: 2017-07-27 13:15:17,407                                                 |                         |                         |                         |                         |                         |                         |                         |
| Total time: 00:00:07,818                                                    |                         |                         |                         |                         |                         |                         |                         |
| Status: Finished                                                            |                         |                         |                         |                         |                         |                         |                         |
| -----                                                                       |                         |                         |                         |                         |                         |                         |                         |
|                                                                             | DataModel               | Catalogs                | Configuration           | AP                      | D1                      |                         |                         |
| F2                                                                          | F3                      | FI                      | ID                      | KP                      | MA                      |                         |                         |
| TD                                                                          | V1                      |                         |                         |                         |                         |                         |                         |
| -----                                                                       |                         |                         |                         |                         |                         |                         |                         |
| Last Sync                                                                   | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 |
| 00:00:00,000                                                                | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 |
| 00:00:00,000                                                                | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 | 0001-01-01 00:00:00,000 |
| Start Time                                                                  | 2017-07-27 13:15:09,605 | 2017-07-27 13:15:09,808 | 2017-07-27 13:15:09,839 | 2017-07-27 13:15:12,667 | 2017-07-27 13:15:12,995 | 2017-07-27 13:15:12,995 | 2017-07-27 13:15:13,884 |
| 13:15:13,884                                                                | 2017-07-27 13:15:14,513 | 2017-07-27 13:15:14,952 | 2017-07-27 13:15:15,242 | 2017-07-27 13:15:15,798 | 2017-07-27 13:15:16,165 | 2017-07-27 13:15:16,165 | 2017-07-27 13:15:17,072 |
| 13:15:17,072                                                                | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 |
| End Time                                                                    | 2017-07-27 13:15:09,808 | 2017-07-27 13:15:09,839 | 2017-07-27 13:15:10,495 | 2017-07-27 13:15:12,682 | 2017-07-27 13:15:12,995 | 2017-07-27 13:15:12,995 | 2017-07-27 13:15:13,884 |
| 13:15:13,884                                                                | 2017-07-27 13:15:14,513 | 2017-07-27 13:15:14,952 | 2017-07-27 13:15:15,242 | 2017-07-27 13:15:15,798 | 2017-07-27 13:15:16,180 | 2017-07-27 13:15:16,180 | 2017-07-27 13:15:17,088 |
| 13:15:17,088                                                                | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 | 2017-07-27 13:15:17,407 |
| Total Time                                                                  | 00:00:00,202            | 00:00:00,031            | 00:00:00,656            | 00:00:00,015            | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            |
| 00:00:00,000                                                                | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            |
| 00:00:00,015                                                                | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            | 00:00:00,000            | 00:00:00,015            | 00:00:00,015            | 00:00:00,015            |
| # Records                                                                   | -                       | -                       | -                       | 0                       | 2                       | 2                       | 0                       |
| 0                                                                           | 0                       | 1                       | 5                       | 1                       | 0                       | 0                       | 0                       |
| 0                                                                           | 0                       | 0                       | 0                       | 0                       | 0                       | 0                       | 0                       |
| -----                                                                       |                         |                         |                         |                         |                         |                         |                         |
|                                                                             | Sync Count              | Max Sync Time           | Min Sync Time           | Avg Sync Time           |                         |                         |                         |
| Current user                                                                | 2                       | 00:00:08,476            | 00:00:07,818            | 00:00:08,147            |                         |                         |                         |
| All users                                                                   | 2                       | 00:00:08,476            | 00:00:07,818            | 00:00:08,147            |                         |                         |                         |

# 2

## Configuring CRM.Client

---

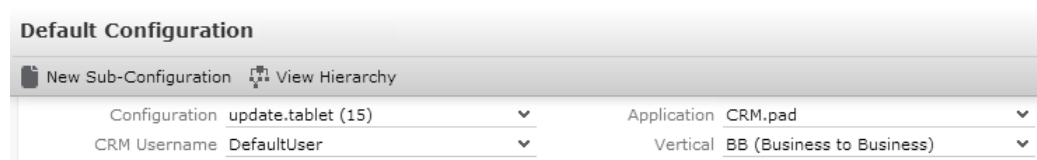
Learn how to configure CRM.Client.

The following topics are available in CRM.Client but are not described in this document. See the Aurea CRM web Administrator Guide for information:

- Defining alternatives, i. e. if you want different icons or pages containing different fields according to the content of a field. For example, appointments can be displayed with a different icon according to its status (scheduled, completed, missed) or an Edit page according to the appointment type (e-mail, letter, etc.).
- [Virtual info areas](#)

## Users and Configurations

The behavior and functionality of CRM.Client is controlled by the current user's configuration and its parent configurations.



| Default Configuration                                          |                           |
|----------------------------------------------------------------|---------------------------|
| <span>New Sub-Configuration</span> <span>View Hierarchy</span> |                           |
| Configuration                                                  | update.tablet (15)        |
| Application                                                    | CRM.pad                   |
| CRM Username                                                   | DefaultUser               |
| Vertical                                                       | BB (Business to Business) |

The default configuration used by CRM.Client and CRM.mobile is called `update.tablet`. This is a root configuration, it is in **no way** connected to `UPDATE_DEFAULT` used for Aurea 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.Client" from the **Application** drop-down list.

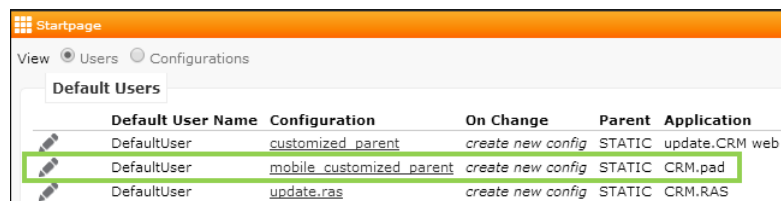
The **Configuration** field is automatically updated to `update.tablet`.

2. Click **New Sub-Configuration**.
3. Enter a name and an (optional) description.
4. Click **Save**.

Since a user cannot change configuration units in CRM.Client 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.

1. If you want to use one custom configuration for all users, assign this configuration to the **DefaultUser**.



| Default User Name | Configuration            | On Change         | Parent | Application    |
|-------------------|--------------------------|-------------------|--------|----------------|
| DefaultUser       | customized_parent        | create new config | STATIC | update.CRM web |
| DefaultUser       | mobile_customized_parent | create new config | STATIC | CRM.pad        |
| DefaultUser       | update_ras               | create new config | STATIC | CRM.RAS        |

To create a user:

1. On the CRM.designer start 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.Client and CRM.mobile)
4. Select "CRM.Client" as **Application**.



|                                   |                                                       |                                            |                                  |                                      |
|-----------------------------------|-------------------------------------------------------|--------------------------------------------|----------------------------------|--------------------------------------|
| <input type="text" value="JOHN"/> | <input type="text" value="mobile_customized_parent"/> | <input type="checkbox"/> create new config | <input type="checkbox"/> dynamic | <input type="text" value="CRM.pad"/> |
|-----------------------------------|-------------------------------------------------------|--------------------------------------------|----------------------------------|--------------------------------------|

5. Click + (**Add**).

When the user logs on to CRM.Client or CRM.mobile, the assigned configuration is used.

## CRM.designer

The configuration tool to configure CRM.Client to your needs is CRM.designer.

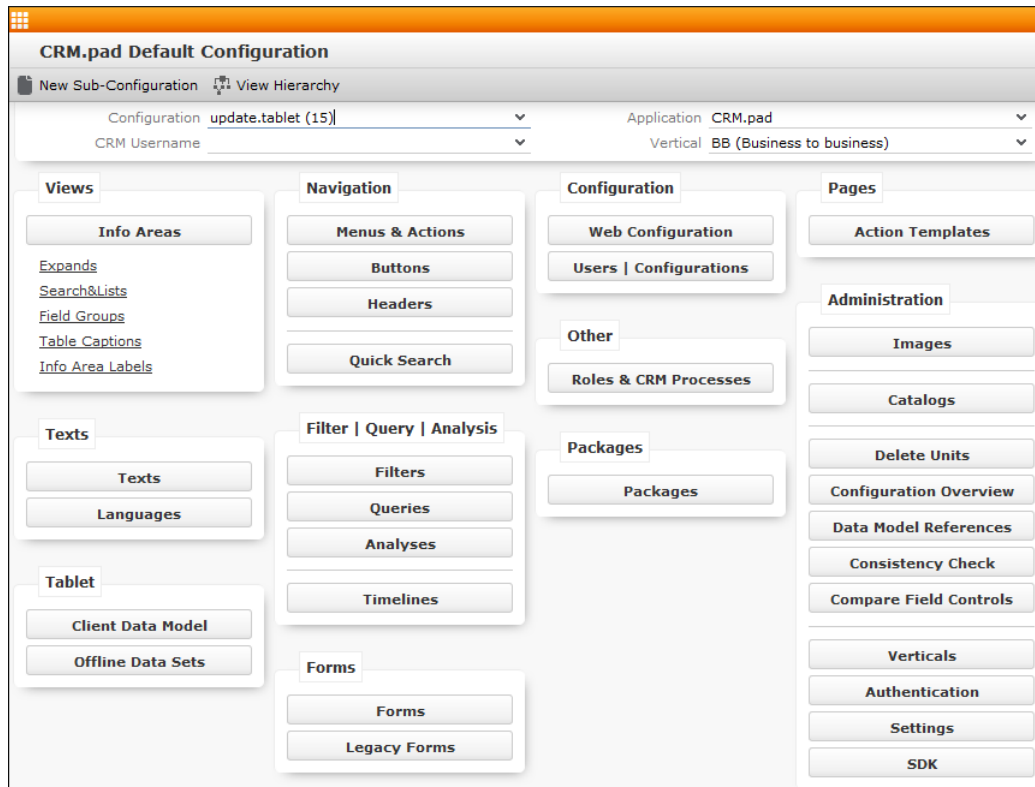
This tool is used to configure all web-based Aurea CRM applications.

Aurea software supplies default configuration called `update.tablet`. This configuration is the base for your configurations. The default configuration is read-only, you cannot change it. You only have access to sub-configurations that you must create.

If you are not used to working with CRM.designer, see the Using CRM.designer topic in the Aurea CRM web Administrator guide.

To call the `update.tablet` configuration:

1. Start CRM.designer. Aurea CRM web's default configuration is displayed on CRM.designer's main page
2. Select `CRM.Client` from the **Application** drop-down list. The default configuration called `update.tablet` is automatically selected from the **Configuration** drop-down list:



The available options are displayed.

**Note:** You must create a sub-configuration, as the default configuration is read-only, see [Users and Configurations](#) on page 18.

## Data Model

Users can access the Aurea CRM database in online mode or a local database in offline mode.

CRM.Client uses a subset of the Aurea CRM data model. You must define this subset in CRM.designer. The following picture shows the default BTB data model for CRM.Client:

| Client Data Model            |        |           |               |                |
|------------------------------|--------|-----------|---------------|----------------|
| Update Data Model            |        |           |               |                |
| Info Area                    | Fields | Relations | Unused Fields | Missing Fields |
| Rep (ID)                     | 21     | 8         | 0             | 0              |
| Company (FI)                 | 33     | 28        | 0             | 0              |
| Person in Company (KP)       | 16     | 19        | 0             | 0              |
| Item Master (AR)             | 0      | 1         | 0             | 0              |
| Questionnaire/Portfolio (F1) | 2      | 9         | 0             | 0              |
| Question (F2)                | 12     | 12        | 0             | 0              |
| Answer (F3)                  | 5      | 10        | 0             | 0              |
| Survey (U1)                  | 2      | 25        | 0             | 0              |
| Survey Result (U2)           | 5      | 11        | 0             | 0              |
| Document (D1)                | 10     | 6         | 0             | 0              |
| Customer Document (D2)       | 7      | 7         | 0             | 0              |
| Opportunity (Y1)             | 15     | 10        | 0             | 0              |
| Activity Participant (MB)    | 5      | 4         | 0             | 0              |
| Favorites (FV)               | 2      | 18        | 0             | 0              |
| Document link (D3)           | 8      | 24        | 0             | 0              |
| Activity (MA)                | 27     | 16        | 0             | 0              |
| Offer (PR)                   | 9      | 11        | 0             | 0              |
| Offer Item (AP)              | 10     | 25        | 0             | 0              |
| Order (AU)                   | 0      | 8         | 0             | 0              |
| Opportunity Product (Y5)     | 6      | 5         | 0             | 0              |

The data model must be updated at least once for each new sub-configuration. The data model displays all info areas used by the selected configuration (and its parent configurations).

To update the data model:

1. Click on **Data Model** on the CRM.designer start page, then click on **Update DataModel**.
2. After updating the data model, data is synchronized automatically when the user logs in.

**Note:** If you add (linked) fields to a field group, you **must** update the data model to include them on the user's device.

## Offline Record Set (Offline Data Set)

You must define an offline data set that contains all the records that are downloaded to the device.

Without this offline data set the rep does not work offline.

A default offline data set is included in CRM.Client. You can adapt it to your needs by adding or removing info areas. You must add a dataset for each info area to your recordset else data cannot be transferred to the rep's device.

Reps can still access records that were not downloaded to the device in online mode.

To define an offline data set:

1. Click **Offline Data Set** on the start page.

| Startpage |                              |                   |
|-----------|------------------------------|-------------------|
| NAME      | INFOAREA                     | CONFIGURATION     |
| AP        | Offer Item (AP)              | 🔍 + update.tablet |
| AR        | Item Master (AR)             | 🔍 + update.tablet |
| D1        | Document (D1)                | 🔍 + update.tablet |
| D3        | Document link (D3)           | 🔍 + update.tablet |
| F1        | Questionnaire/Portfolio (F1) | 🔍 + update.tablet |
| F2        | Question (F2)                | 🔍 + update.tablet |
| F3        | Answer (F3)                  | 🔍 + update.tablet |
| FI        | Company (FI)                 | 🔍 + update.tablet |
| ID        | Rep (ID)                     | 🔍 + update.tablet |
| KP        | Person in Company (KP)       | 🔍 + update.tablet |
| MA        | Activity (MA)                | 🔍 + update.tablet |
| MB        | Activity Participant (MB)    | 🔍 + update.tablet |
| PR        | Offer (PR)                   | 🔍 + update.tablet |
| Y1        | Opportunity (Y1)             | 🔍 + update.tablet |
|           | Language (00)                | +                 |

2. For each offline data set, you can define:

- The info area.
- A list of filters. Only the records meeting the filter criteria are available offline.
- A list of related datasets. Records included in this dataset can be linked offline.

3. When you add a dataset, you can select a filter.

Startpage > Offline Data Sets

**InfoArea:** Activity (MA)

**DisplayText:** +

**Ignore:** ☐

**Recordsets:**

| TYPE     | NAME                   |
|----------|------------------------|
| none.    |                        |
| + Filter | MA.ChildCreateTemplate |

**Sync Dependent:**

| DATASETNAME | INFOAREA               | LINKID | FILTER    |
|-------------|------------------------|--------|-----------|
| X FI        | Company (FI)           |        |           |
| X KP        | Person in Company (KP) |        |           |
| + AP        |                        |        | no filter |

Save

4. You can disable datasets with the **Ignore** flag.

All defined fields and links that were defined in the **Infoareas** section are synchronized for each record.

A record is synchronized if it matches at least one of the filters in the datasets OR is linked to a record contained in at least one of the related datasets listed in the **Sync Dependent** area (only filters of the related datasets are applied, cascading links are not supported) and also matches the filter if defined.

If a dataset contains (Lookup) in its name, these records are also synchronized, but they are stored for reference purposes only. The data is shown wherever it was added as linked fields, but is not displayed in Details view or in search results.

## Offline Record Set (Offline Data Set) - Example

You can define that only the companies and persons of a rep's Favorites are available on the device.

In addition, you want persons (not defined as Favorites) of companies in the Favorites, and companies (not defined as Favorites) of persons in the Favorites to also be transferred.

To configure this:

1. Define the following `FI.Favorites` filter:

Startpage Filter

**Configuration for Filter 'FI.Favorites'**

Display-Text: +

New-Date: 22.08.2013 15:57:21

☐ Readonly

☐ Invisible in lists (within the application)

Description: Enter description here.

**Company (FI)** + ×

Filter: FI-StaNo func: + +FIELD

**Favorites (FV)** + × ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: + × Rep ID = \$CurRep × + ( ) Parameter +

Save

This filter identifies the company records that were added to the Favorites.

2. Define the following `FI.FIKP.FAV` filter:

Startpage Filter

**Configuration for Filter 'FI.FIKP.FAV'**

Display-Text: +

New-Date: 22.08.2013 16:05:56

Upd-Date: 22.08.2013 14:12:32

☐ Readonly

☐ Invisible in lists (within the application)

Description: Enter description here.

**Company (FI)** + ×

Filter: + × Rep 1 ID = + ( ) Parameter +

**Person in Company (KP)** + × ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: Postal Code func: + +FIELD

**Favorites (FV)** + × ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: + × Rep ID = \$curRep × + ( ) Parameter +

Save

This filter identifies the companies (not defined as Favorites) that are linked to a person in the Favorites.

3. Define the following dataset for FI:

Startpage > Offline Data Sets

**InfoArea: Company (FI)**

DisplayText:

Ignore: ☐

**Recordsets:**

|  | TYPE   | NAME              |
|--|--------|-------------------|
|  | Filter | FI.Favorites      |
|  | Filter | FI.FIKP.FAV       |
|  | Filter | FI.BasedInAustria |

**Sync Dependent:**

| DATASETNAME | INFOAREA | LINKID | FILTER    |
|-------------|----------|--------|-----------|
| none.       |          |        |           |
|             | AP       |        | no filter |

Save

4. Define the following `KP.Fav` filter:

Startpage > Filter

**Configuration for Filter 'KP.FAV'**

Display-Text:

New-Date: 22.08.2013 16:20:29

☐ Readonly

☐ Invisible in lists (within the application)

Description: Enter description here.

**Person in Company (KP)**

Filter: Postal Code = ( ) Parameter

**Company (FI)** ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: Rep 1 opt. = ( ) Parameter

**Favorites (FV)** ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: Rep ID = \$curRep ( ) Parameter

Save

This filter identifies the company records that were added to the Favorites.

5. Define the following `KP.KPFI.FAV` filter:

Startpage > Filter

**Configuration for Filter 'KP.KPFI.FAV'**

Display-Text: +

New-Date: 22.08.2013 16:27:49

☐ Readonly

☐ Invisible in lists (within the application)

Description: Enter description here.

**Person in Company (KP)** + ×

Filter: + × Postal Code = + ( ) Parameter +

**Company (FI)** + × ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: + × Rep 1 opt. = + ( ) Parameter +

**Favorites (FV)** + × ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: + × Rep ID = \$curRep × + ( ) Parameter +

Save

This filter identifies the persons (not defined as Favorites) that are linked to a company in the Favorites.

6. Define the following dataset for KP:

Startpage > Filter

**Configuration for Filter 'KP.KPFI.FAV'**

Display-Text: +

New-Date: 22.08.2013 16:27:49

☐ Readonly

☐ Invisible in lists (within the application)

Description: Enter description here.

**Person in Company (KP)** + ×

Filter: + × Postal Code = + ( ) Parameter +

**Company (FI)** + × ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: + × Rep 1 opt. = + ( ) Parameter +

**Favorites (FV)** + × ☐ LinkId: Relation: WITH MaxRec: 0 ☐ Optional

Filter: + × Rep ID = \$curRep × + ( ) Parameter +

Save

7. Furthermore, define the following Dataset for MA if you want to transfer the appointments for these companies or persons:

Startpage > Offline Data Sets

**InfoArea: Activity (MA)**

**DisplayText:**

**Ignore:** ☐

**Recordsets:**

| TYPE                                    | NAME                   |
|-----------------------------------------|------------------------|
| none.                                   |                        |
| <input type="button" value="⊕"/> Filter | MA.ChildCreateTemplate |

**Sync Dependent:**

| DATASETNAME                         | INFOAREA               | LINKID | FILTER    |
|-------------------------------------|------------------------|--------|-----------|
| <input type="button" value="⊗"/> FI | Company (FI)           |        |           |
| <input type="button" value="⊗"/> KP | Person in Company (KP) |        |           |
| <input type="button" value="⊕"/> AP |                        |        | no filter |

## Using JSON

You can define parameters using JSON syntax.

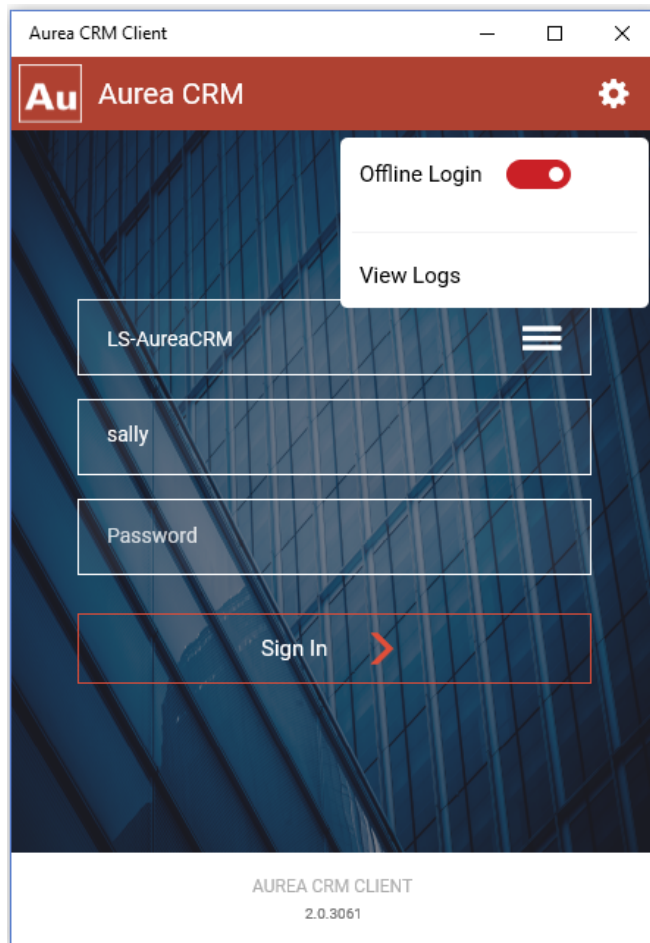
These JSON strings must be valid JSON as determined by the iOS or Windows 10 parser, for example,

- Must start and end with double quotes
- Each element name must be placed in double quotes
- Double quotes in strings must not be escaped

Example for a valid JSON string: {"MaxResults": "5"}

## Login Page

The login page allows the user to log on to the application.



The user can select the following options in the  (**Settings**):

- Switch between online and offline Mode
- View the log file and send it to the administrator

You cannot add or remove items from the **Settings**. However, you can:

- Add a background image and/or a logo, see [Adding a Logo or Background Image to the Login Page](#).
- Add an Insight Board (included in the `update.tablet` configuration) that, for example, contains information on the application, a demo or a tutorial, see [Adding an Insight Board to the Login Page](#) on page 29.

## Adding an Insight Board to the Login Page

You can define that an Insight Board is also displayed on the login page.

The items assigned to this Insight Board must have an alternative context menu assigned (`ContextMenu` argument in the `InsightBoardItem` action call, see [Insight-BoardItem](#) on page 80).

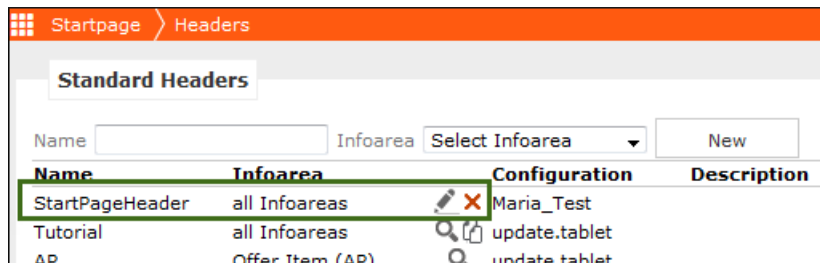
**Note:** The alternative context menu is restricted to items having the [WebView](#) on page 89 action assigned with the `ReportType` argument set to "Url".

## Adding Buttons to the Start Page

Learn how to add buttons to start page.

To add buttons to the start page:

1. Click **Headers** on the CRM.designer start page.
2. Add a header group that applies to all info areas, for example, `StartPageHeaders`.



3. Switch to this header group and add a **Special Header**, for example, `StartPage-Header`.
4. Add the desired buttons to this header. You can only add buttons without a record context, for example, adding new records like tasks or activities.
5. Enter this special header's name in the `HeaderName` input argument of the `DashboardView` action call, [DashboardView](#) on page 79. This action is called in the default configuration by the `DASHBOARD_STARTPAGE` application menu action.

Syntax:

```
<Header Group Name>.<Special Header Name>
```

Example:

```
StartPageHeader.StartPageHeader:
```

Action for Menu DASHBOARD\_STARTPAGE

Action Template: **DashboardView**

☐ Pass arguments as an object ?

Input Arguments:

|                   |                                        |
|-------------------|----------------------------------------|
| ConfigName        | StartPage                              |
| ConfigTabNr       |                                        |
| <b>HeaderName</b> | <b>StartpageHeader.StartpageHeader</b> |
| RecordId          | <No Value>                             |

+ New Argumentname

## Menus

CRM.Client navigation menu is configured in the **Menus & Action** area in the `update.tablet` configuration.

The user can navigate in CRM.Client using the  (**Search**) menu. See [Configuring the Search Menu](#) on how to define the Search menu.

### User-defined Menus

The user can navigate through CRM.Client using the  (**Search**) menu. See [Configuring the Search Menu](#) on how to define the Search menu.

## Buttons

You can define buttons to pages that allow the user to call the assigned page.

You can add the following buttons to the Details (or any other) view:

- **Edit**: Calls `OrganizerAction` action template, see [OrganizerAction](#) on page 83.
- **EditWithSerialEntryButton**: Call the quick add, see [SerialEntryEdit](#) and [SerialEntryNew](#).
- **DownloadRecord**: Calls the `SyncRecord` action template, see [SyncRecord](#).
- **Modify** (not included in the `update.tablet`): You can add this button if you want to change records based on an existing Template filter by assigning the `ModifyRecord` action call, see [Template Filter](#) and [ModifyRecord](#).

---

To add a modify button, configure the `ModifyRecord` page (which actually is a function) to hand over the record and to apply a `TemplateFilter`.

- `Delete`: Calls the `Delete` organizer action to delete the current record, see [OrganizerAction](#) on page 83.

## Adding a Button

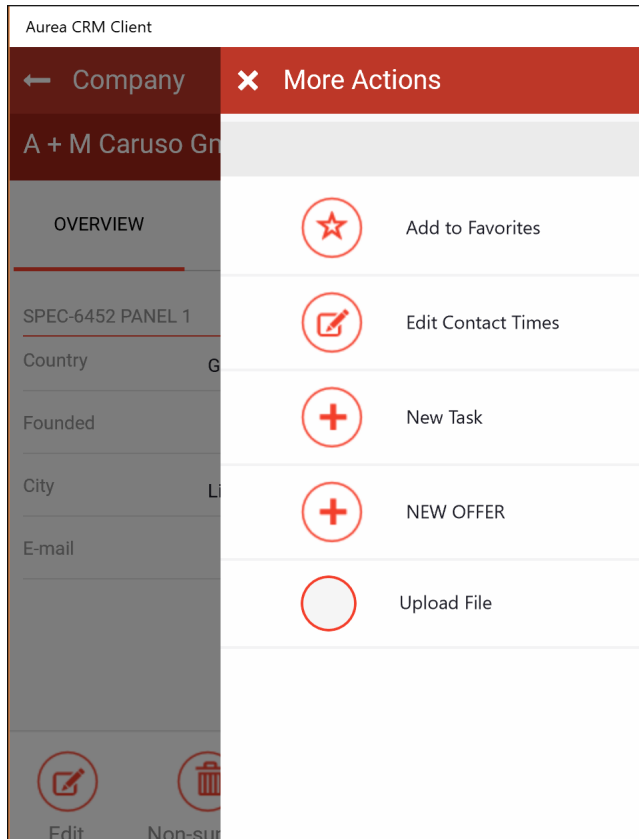
Buttons are added to the header group of the desired info area for each view.

To add a button to the Details (or any other) view:

1. Switch to the header group (for example, Expand) of the desired info area (for example, Person).
2. Select a button in the drop-down list at the end of the existing buttons and click on  (**Add**).
3. Click on the button's name to open the button's details.
4. Click on **Copy to this configuration** to switch to the Edit mode, then click on  (**Edit**) besides **Action** to assign the action that is executed when the user clicks on this button.
5. Define the action's parameters. The mandatory parameters depend on the selected button.

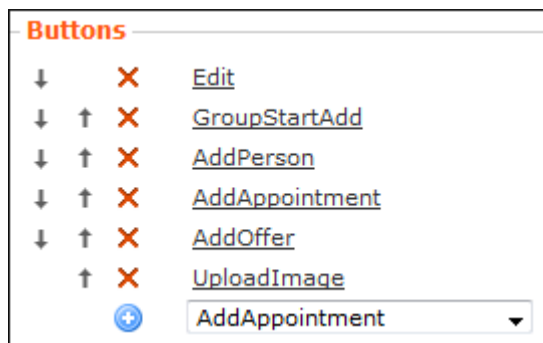
## Adding a Button Group

You can also define that one or more buttons are grouped and that the options are displayed in a pop-up list when the user clicks the **More Actions** menu in the record's header:



To add a button group:

1. Switch to the header group for the desired info area, for example, Company.
2. Add the desired buttons to the **Buttons** area:



3. Add the `GroupStart` button (or any button with a name starting with `GroupStart`). The image defined for this button is used as icon for the button group.
4. Add the `GroupEnd` button to mark the end of a group. The `GroupEnd` button is only required if you want to add further buttons at the right side of a group.

**Note:** The grouping is ignored if quick actions are not defined.

## Control Center

The control center contains functions that the user can call throughout the application such as the synchronization options or the system information.

You can define global actions that are available in the **Global** area of the control center, see [Global Actions \(not supported in this release\)](#).

---

**Attention:** Global actions are not supported in this release of CRM.Client. The Global area of the control center is empty.

---

In the control center the user can:

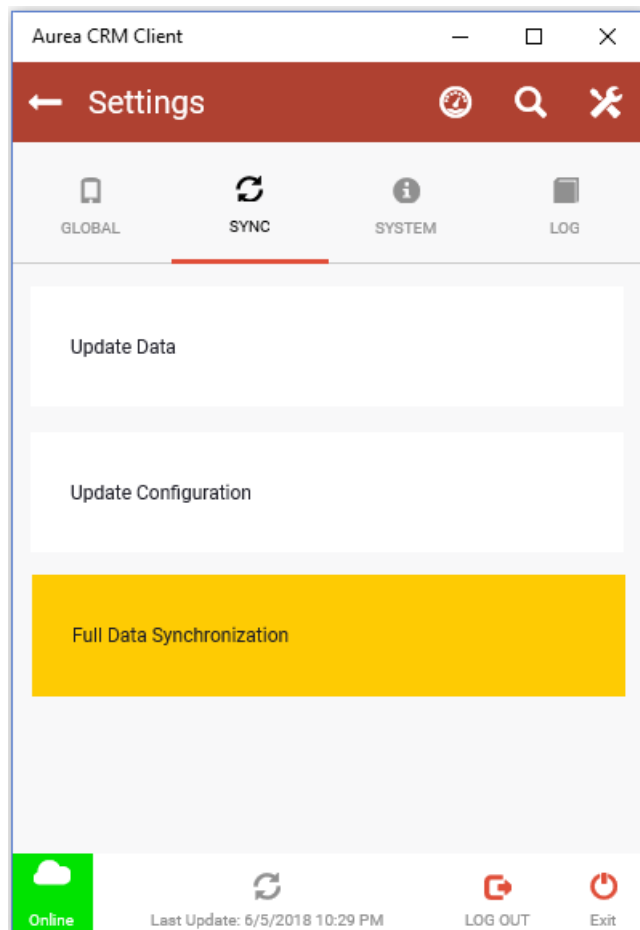
- See, which mode (online or offline) is currently active.
- Log out from the application
- Switch between the options: **Global**; **Sync**; **System**; **Log**.
- Close the CRM.Client application.

## Synchronization

The user can synchronize data in the **Sync** area of the control center.

The user can synchronize either:

- All data, which includes the data and the configuration, see [Full Data Synchronization](#).
- Only the data that has been changed since the last synchronization, see [Update Data](#).
- Only the configuration (all the data from the designer database including resources), see [Update Configuration](#).
- Change the application's language.



---

**Note:** Data is not synchronized when the device switches to sleep mode, synchronization processes are interrupted (even if a full synchronization is being carried out). Therefore, Aurea recommends that you delay the switch to sleep mode until full synchronization is completed.

---

You can use the following Web Configuration parameters to disable options:

- [Sync.ConfigOnOff](#).
- [Sync.LanguageOnOff](#).

### Full Data Synchronization

The full data synchronization transfers the data model, catalog values, the configuration, record data, configuration resource files, and documents as defined beforehand.

---

**Note:** Records deleted on the server are not transferred to the device (are not deleted on the device) unless a full synchronization is carried out.

---

Data is transferred in several requests that are sequentially processed:

- At first, the data model, catalog and configuration data is transferred from the server to the client.
- A request for each offline data set fetches the record data.
- All configured resource files are transferred.
- All document requests are sent to the Document Download Manager that downloads them asynchronously.

The application is blocked during synchronization, other actions cannot be carried out.

Data is fully synchronized when a user accesses the application for the first time. Afterward, the user is reminded to fully synchronize data regularly. After a certain time period, users cannot access the application anymore without carrying out a full synchronization. You can define the interval when a user must carry out a full synchronization with the following web configuration parameters:

- [Sync.FullSyncWarnIntervalDays](#).
- [Sync.FullSyncIntervalDays](#).
- [Sync.FullSyncBlockIntervalDays](#).

If the application is terminated during the synchronization, the state is stored on the device. If the user logs in again, the synchronization continues.

---

**Note:** Records that are not included in the offline dataset are removed from the device when the next full synchronization is carried out, see [Offline Record Set \(Offline Data Set\)](#) on page 22.

---

## Update Data

Data synchronization transfers all records and catalog data changed since the last synchronization (full or incremental). It is executed in one request, usually only takes seconds and can be carried out frequently.

You can define that data is incrementally synchronized at specific time intervals, see [Sync.Config](#).

After synchronizing data is up-to-date, **except**:

- Data downloaded with the application but not belonging to an existing record set.
- Data that does not belong to an offline data set anymore because the rights definition or the record set itself was changed.
- Data belonging to an offline data set after the rights definition was changed but the data itself was not changed. This data is only included in a full synchronization.

---

**Note:** Records that have been marked as deleted (DelCd. field) in the Aurea CRM database are also synchronized. However, physically deleted records are not considered. They are only removed from the device when carrying out a full synchronization.

---

## Update Configuration

It is recommended to use this option only during the project development phase. Only the application's configuration data is transferred.

## UpSync

The UpSync is started automatically when the application switches from offline to online mode and synchronizes only requests that were created while working offline.

## System Information

System information is available in the control center's **System** area.

The following configuration units are included in the `update.tablet` configuration:

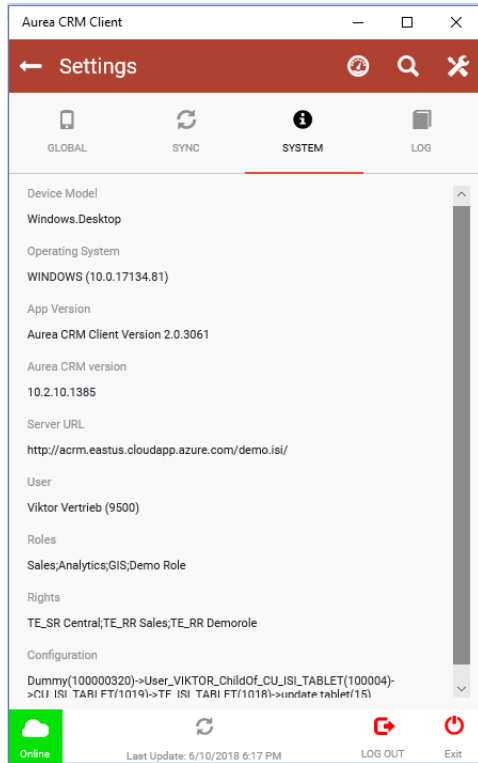
- Context menu action called `INFO_SYSTEMINFO`. This context menu action calls the `SystemInfoPage` action, see [SystemInfoPage](#) on page 89.
- Company (FI) info area:
  - Field group called `SYSTEMINFO`
  - Header group called `SYSTEMINFO`

---

**Note:** This header group does not apply to version 2.x but exists for compatibility purposes for earlier versions.

---

The **System** area gives an overview of information on the current set-up (such as the version number, server URL, assigned rights, logged-in user, configuration, etc.) and facilitates user support.



You can add additional information as follows:

1. Create a **Web Configuration Layout** named `SystemInfo`. Every parameter in this layout is automatically appended at the end with its name. See the [Aurea CRM web Administrator Guide](#) on how to create a **Web Configuration Layout**.

---

**Note:** You cannot change modify but only display these additional parameters.

---

2. If you want a different label displayed for a parameter, add an additional (dummy) field to the `SYSTEMINFO` field group, enter the name of the desired Web Configuration Layout parameter in the **Function** field and define the **Label**.

### Default SYSTEMINFO Field Group

The data displayed in the System Information is defined in the List control of the FI info area's `SYSTEMINFO` field group. The default is configured as follows:

Startpage > All Info Areas > Info Area FI > Field Group SYSTEMINFO

Field-Group 'SYSTEMINFO', Control 'List'

Copy to configuration

0/S - FI-StaNo  
1/L - FI-SerNo  
2/C - Company  
3/C - Synonym  
4/C - Division  
5/K - Country  
6/C - County

**Fields**

FreeC1 (60) FreeC2 (61) FreeC3 (62) FreeC4 (147)  
FreeC9 (152) FreeC5 (148) FreeC6 (149) FreeC7 (150)  
FreeC8 (151)

**Field "FreeC1"**

XML "FreeC1", Field Nr. 60, Type C

Label: Device Model  
Tool Tip:  
Function: devicemodel

These fields are used as placeholders to assign **Function** names to them. Ensure that the following data is given in the **Label** and **Function** fields for each field:

**Note:** You can use any of the field values, and use your own labels. The **Field** and **Label** names provided in the table below are only samples. But the function names cannot be changed and each function returns specific system information.

| Field  | Label                 | Function         |
|--------|-----------------------|------------------|
| FreeC1 | Device Model          | devicemodel      |
| FreeC2 | Operation System      | system           |
| FreeC3 | App Version           | appversion       |
| FreeC4 | Aurea CRM web Version | webversion       |
| FreeC5 | Server URL            | serverurl        |
| FreeC6 | User                  | user             |
| FreeC7 | Roles                 | roles            |
| FreeC8 | Rights                | rights           |
| FreeC9 | Configuration         | configuration    |
| FreeK1 | Device Model          | rasapplicationid |

**Note:** If you are using a RAS application then you can display the RAS application ID by using the `rasapplicationid` function.

## Log Files

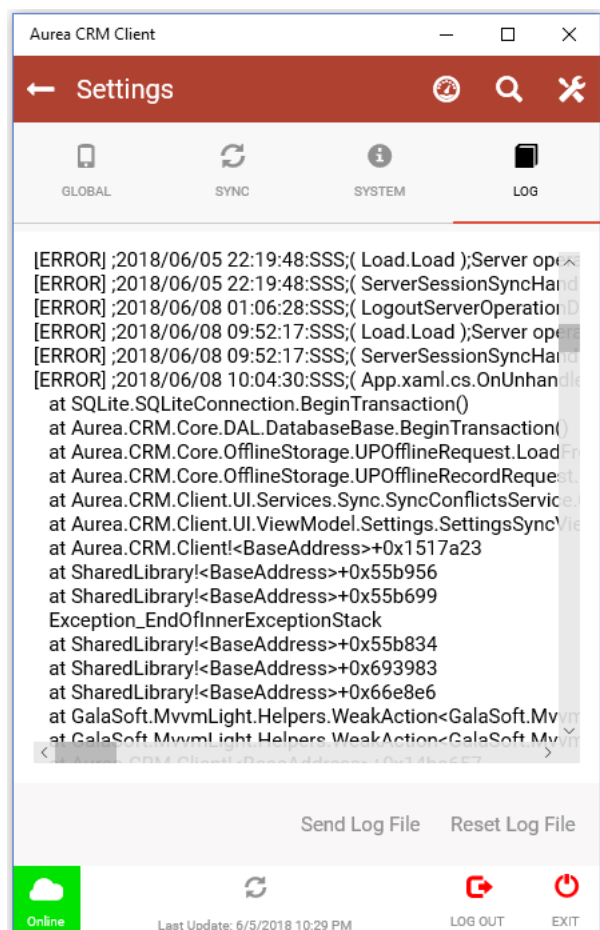
Logging information is displayed in the **Log** area and can be sent by e-mail to your administrator.

Logging is enabled by default after installing CRM.Client, the log level is automatically set to `Warning`.

Log files can be viewed as follows:

- In the **Log** area of the control center. You can send the log file by e-mail, reset the log file or query the available databases (SQL statements).
- If login was not successful, by tapping on  (**Settings**) and selecting **View Logs**.

The content of the log file is displayed:



The user can reset the logged information content (**Reset Log File**) or send it by e-mail (**Send Log File**).

You can define the following behaviour using the Web Configuration parameters:

- When an error occurs, a button is displayed in the pages with which the user can send the e-mail, see [SendErrorReport](#).
- Additional information such as screenshots etc. is included in the sent data, see [SendErrorContextInfo](#).

---

**Note:** During login, if the log file size exceeds 10MB then the last 5MB of log information is retained and the remaining purged. A notification message pops up notifying you about the same.

---

## Overruling Default Log Settings in the CRM.Client

In addition to the **Log** area in the CRM.Client's control center, more logging options are available for CRM.Client users in iOS and Windows 10 devices. These options are available for each user and can help detect problems and troubleshoot issues. For example, the **Support Center** can ask the user experiencing problems to change the log level (for example, to track a specific error or when no access to the CRM.designer database is available) and then to e-mail the log file to the Support Center agent.

Only set these options for testing purposes or to locate the cause of errors, as each enabled item increases the data transfer and, therefore, can lead to performance problems.

The iOS device user must switch to the device settings () and tap on **CRM.Client** icon to enable these additional settings.

Windows Desktop and Mobile users can configure these settings in the `log-settings.json` file located at `<Drive Name>:\Users\<user>\AppData\Local\Packages\178E3AFA.AureaCRMClient_*****\LocalState`. Where `<user>` is the name of the user configured on the Windows 10 device, and `<Drive Name>` for example is `C` or `D`.

The following parameters can be configured for Logging Information:

---

**Note:** These settings overrides the corresponding configuration in the web configuration log parameters, see [Log Parameters](#) on page 90.

---

- **Override Server Log Level:** If set to `ON`, only options set to `ON` of the available log options are logged. (default: `OFF`)
- **Log Level:** Defines which occurrences are written to the log file. Takes the log levels values from 0 to 3, where: 0-None; 1-Error (Default); 2-Warning; 3-Info. Overrides the settings in the web configuration parameter [Log.Level](#).
- **SQL Log:** SQL statements of the local database are logged. (default: `OFF`)
- **Request Log:** Server requests (sent and received) are logged. (default: `OFF`)
- **Config Log:** Configuration items (loaded or used) are logged. (default: `OFF`)
- **Network Log:** General network communication is logged. (default: `ON`)

- **Result Log:** Each result row is logged. (default: OFF)
- **UpSync Log:** UpSync requests are logged. (default: OFF)

**Note:** ON = true. OFF = false

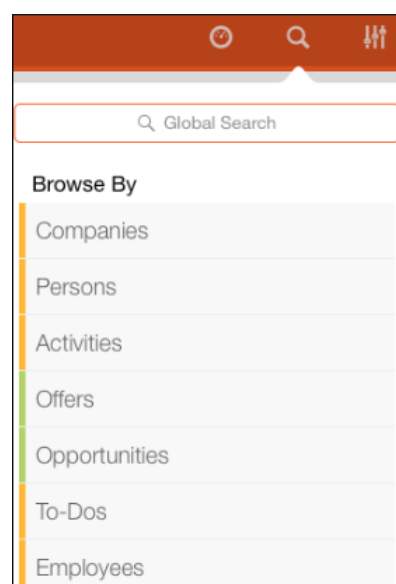
The following sample shows the `log-settings.json` configuration settings for Windows 10.

```
{ "OverrideServerSettings":false, "LogLevel":1, "LogClass":"","LogConfig":false,
  "LogMethod":"","LogNetwork":true, "LogRequests":false, "LogResults":false,
  "LogStatements":false, "LogUpSync":false}
```

## Search

The Search area contains the global search as well as all available info area-related searches from the SMARTBOOK menu by default.

The user can open the Search area by clicking on the  button. The Search button is available throughout the application.



Find as you type is activated for all searches by default. The user can tap on the desired record in the search result to switch to the record's details view.

### Configuring the Search Menu

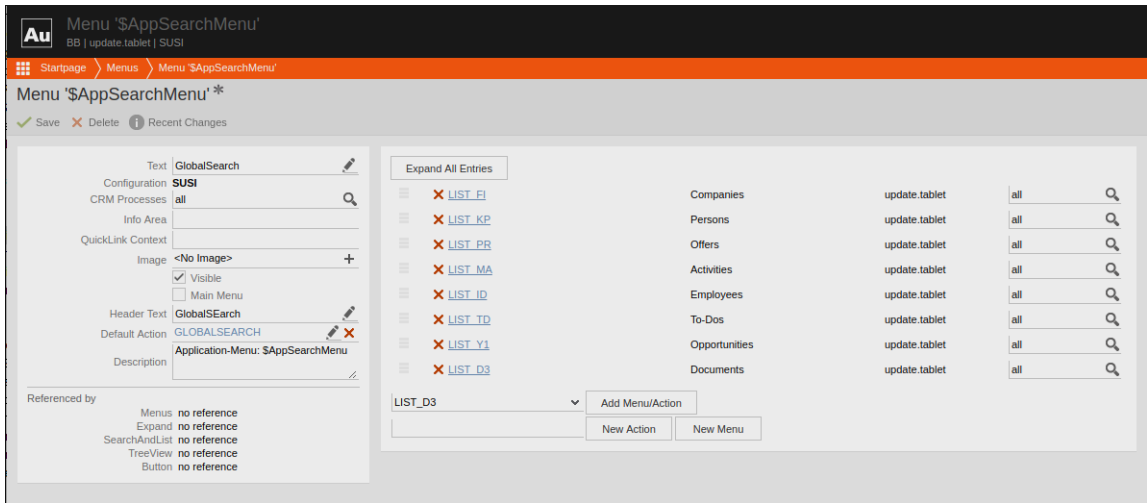
In CRM.Client you need to define the info area-related searches in an application menu named `$AppSearchMenu`.

To define your own search menu:

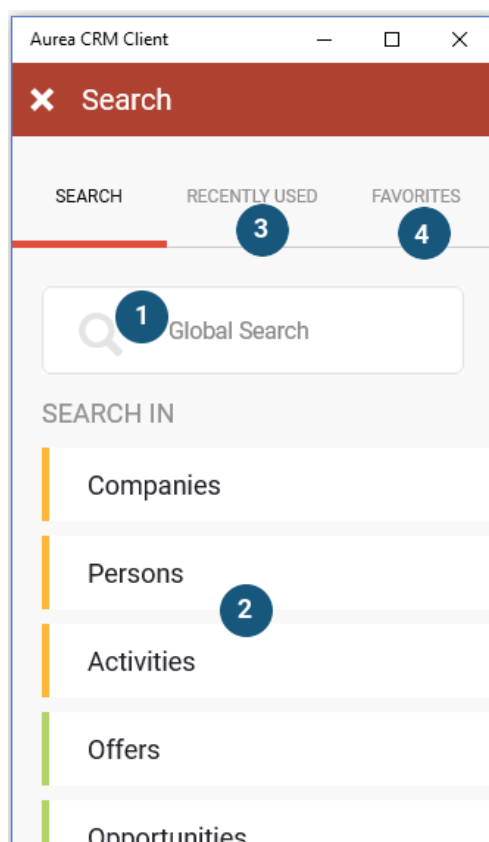
1. Add an application menu named `$AppSearchMenu`.
2. Add the desired items and assign the following action templates.

| Application Menu Action Name                  | Action Template                                |
|-----------------------------------------------|------------------------------------------------|
| <code>\$GlobalSearch</code>                   | see <a href="#">GlobalSearch</a> .             |
| For each info area an application menu action | see <a href="#">RecordListView</a> on page 85. |

The configuration is shown in the screenshot below:



The resulting menu can consist of the following items:



|   |                                                                                                                                                                                             |
|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | The global search: Data is searched in all info areas on the device (offline mode).                                                                                                         |
| 2 | Info area-related searches: You must create a context menu action with the <code>RecordListView</code> action template assigned for each info area. All other action templates are ignored. |
| 3 | History search: The recently viewed records are listed.<br><b>Attention: History Search</b> is not supported in this release of CRM.Client.                                                 |
| 4 | Favorites: The records you defined as Favorites are listed.<br><b>Attention: Favourites</b> is not supported in this release of CRM.Client.                                                 |

## Searching Several Info Areas (Global Search)

The Global Search is included in the default configuration. This search includes more than one info area. You can define the info areas to be searched in the **Quick Search** area. The user can access the Global Search from the application menu.

To define the Global Search:

1. Click on **Quick Search** on the CRM.designer start page:

| Search Name       | Fieldname  | Tablename              | Processes | Alternate Menuaction Name                 |
|-------------------|------------|------------------------|-----------|-------------------------------------------|
| Company Search    | Company    | Company (FI)           | all       | <input type="checkbox"/> View Field Label |
| Last Name Search  | Last Name  | Person in Company (KP) | all       | <input type="checkbox"/> View Field Label |
| First Name Search | First Name | Person in Company (KP) | all       | <input type="checkbox"/> View Field Label |
| Rep               | Rep        | Activity (MA)          | all       | <input type="checkbox"/> View Field Label |
| Rep ID            | Rep ID     | Activity (MA)          | all       | <input type="checkbox"/> View Field Label |

Buttons: Add new Search, Save, Delete

2. Define the fields in the info areas to be searched:

- Click on (**Edit**) besides an item to change the existing setting.
- Click on (**Add**) at the end of the list to add a new info area, and then define the settings for the search.

**Note:** The `FullTextSearch` web configuration parameter does not apply to the global search for performance reasons.

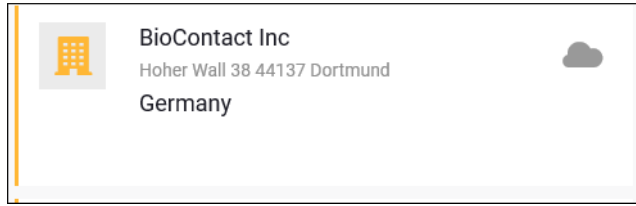
## Configuring the Search&List Page for Info Area Searches

After searching for data, the hits are listed in the Search&List page. For the Search&List page define the following configuration units for the desired info areas in CRM.designer:

- Search control: Defines the fields that are searched in the database. These fields are shown in the application's search field as ghost text.

- List control: Defines the data displayed in lists.

You can also define that an icon is displayed for the info area. Assign an image with the same name as the info area ID (for example, FI for company) in the Expand Page configuration:



- Search header: The header is defined in the header group and then assigned in the Search&List configuration. You can add to a header group:
  - The page's title (header text, defined in the Text field).
  - Buttons: If you add buttons, the first three buttons are displayed as quick actions on the left side of the header, the other buttons can be accessed by the **Actions** menu on the right side.

## Displaying Record Details

Learn how a record's details are displayed is defined in the `RecordView` page.

You can add a `RecordView` page only to a record's context-specific menu action in the **Context Menus** area. The `RecordView` page is usually called by the predefined `SHOWRECORD` menu action that displays the record clicked in a list with the default Expand Page Configuration.

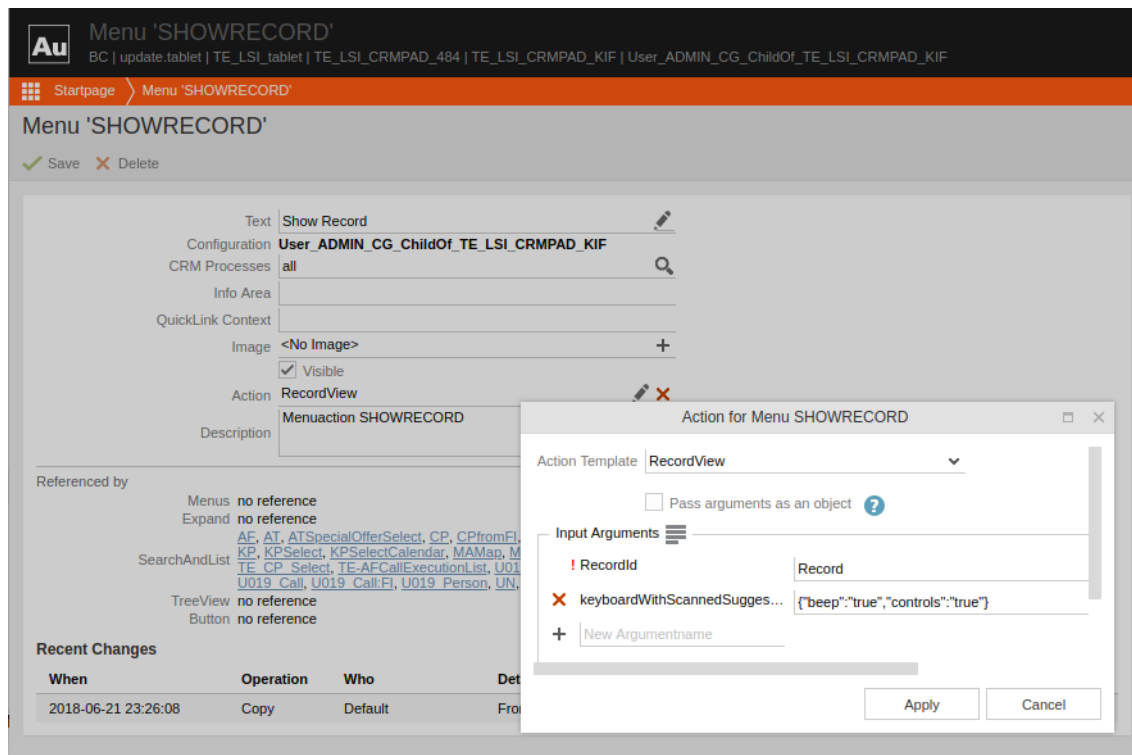
The Expand configuration defines which field group and header group is used.

To define the Details page:

1. Switch to the **Context Menu Actions** area.
2. Select `SHOWRECORD` and copy it to your configuration.

The menu's configuration is displayed.

3. Click on  (**Edit**) besides **Action**.



## Displaying Data in Additional Tabs

You can define that linked data is available in CRM.Client in additional tabs on the right side of the screen.

You can add additional tabs to lists or to Expand views. For example, in Expand view they can contain child info areas, and in lists, as tab could contain the current user's appointments.

To add additional tabs to display data:

1. Switch to the header group of the desired info area. The following example shows the Person (KP) info area:

| Infoareas           |               |          |            |        |         |          |           |               |              |        |           |       |
|---------------------|---------------|----------|------------|--------|---------|----------|-----------|---------------|--------------|--------|-----------|-------|
| Order               | InfoArea Link | CRM Proc | Fieldgroup | Header | Text    | Filter   | Tab Style | SubListParams | Menu         | Search | Form Name | Flags |
| ↓                   | Activity (MA) | all      | MA         | MA     | Default | no value |           |               |              | MA     |           | 0     |
| ↓                   | Offer (PR)    | all      | PR         | PR     | Default | no value |           |               |              | PR     |           | 0     |
| ↑                   | Company (FI)  | all      | FI         | FI     | Default | no value |           |               | SHOWFIRECORD | FI     |           | 0     |
| Rep (ID) All tables |               |          |            |        |         |          |           |               |              |        |           |       |

The Person Details view contains three additional tabs:

- The first tab contains the person's activities (a search page with a link record).
  - The second one contains the person's offers (a search page with a link record).
  - The third one displays the page that was configured in the `SHOWFIRECORD` menu action. You can add this context menu action on any page. This `SHOWFIRECORD` page displays a Details view with the context record as link record and a fix target info area.
2. You can enable **Link** if the info area is not linked to the current info area with the default link.
  3. Enter an existing context menu action in the **Menu** field to call the defined action when the tab is displayed.

---

**Note:** You can add a filter to the tabs. Define a context menu action that calls the desired filter and enter this context menu action's name in the **Menu** column.

---

4. Enter the text that is displayed on the tab in the **Text** column
5. To change the text of the first tab, add a sub-info area row on top and set **TabStyle** to "#" in that row.

---

**Note:** **Fieldgroup** and **Headergroup** are not used. These groups are defined by the **Search** field.

---

## Info Areas

Learn about CRM.Client info areas.

## Field Groups

Learn about the fields group info areas.

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

## Field Controls

Learn about the field controls info area for CRM.Client.

The following field controls are used by CRM.Client.

### Details Control

You can define the contents of the Details view in the Details control. In addition, it is used as a fallback if no Edit control was defined.

As in Aurea CRM web, fields are grouped in panels, see [Panels](#) on page 54. Each panel has its own label and panel separator.

Each field has its own label, field value, and separator. If a field value is empty, then the field is not displayed in the Details view.

If no child records are found on the device (the database on the device does not contain any records for an info area), CRM.Client automatically searches for data online.

### Defining Field Contents to be shown in the Details Header

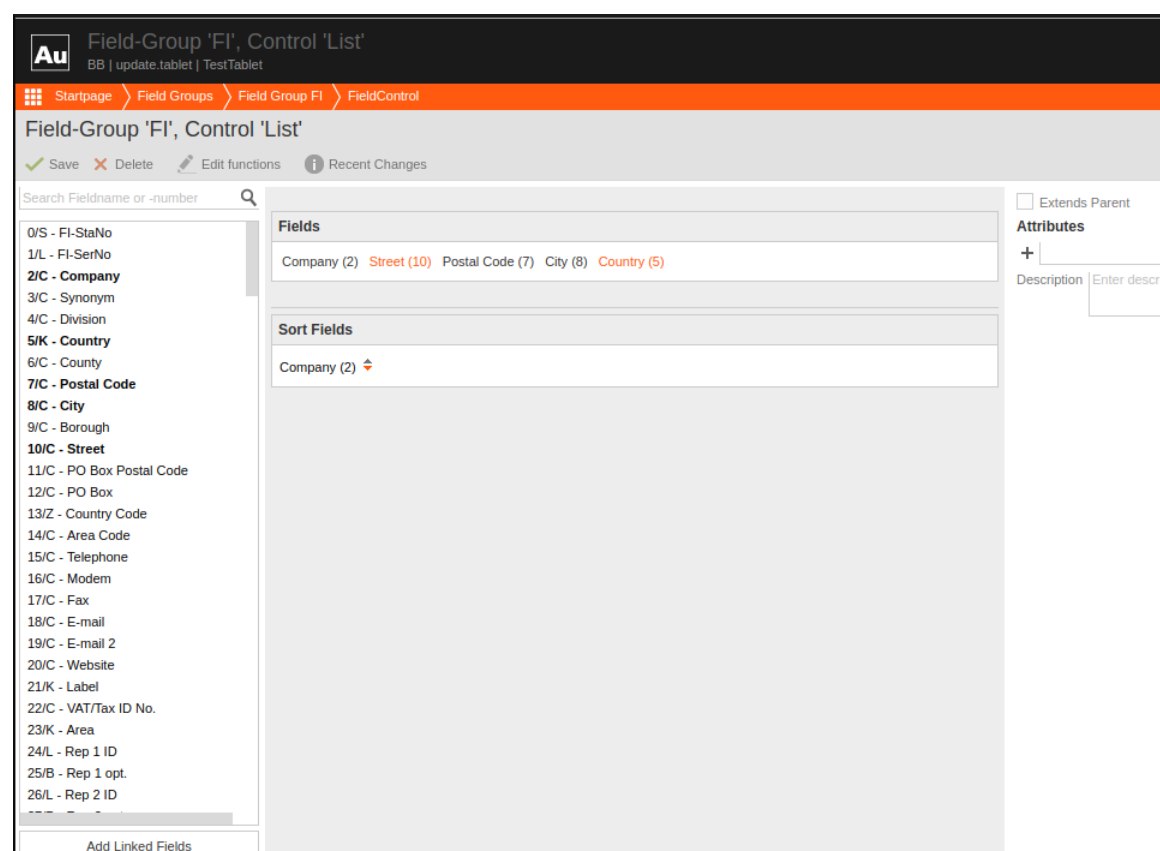
You can define in the Details control that fields from the Details view are displayed in the header below the table caption.

Assign `OrganizerHeaderSubLabel` as **Function** to the field(s) you want to display in the header. You can also assign the `Colspan` field attribute, see [Field Attributes](#) on page 50.

### List Control

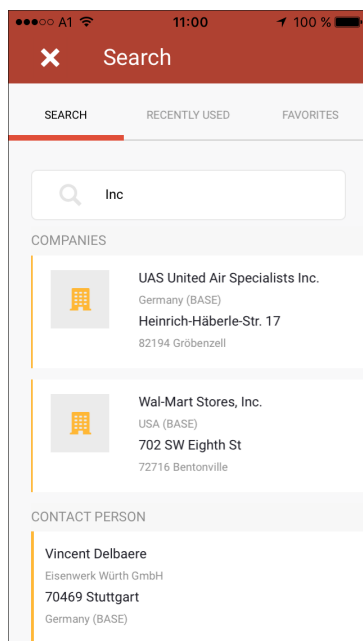
You can control the Fields of a record displayed in a list and the Sort Field used to sort the records displayed in the list.

See the screenshot below:

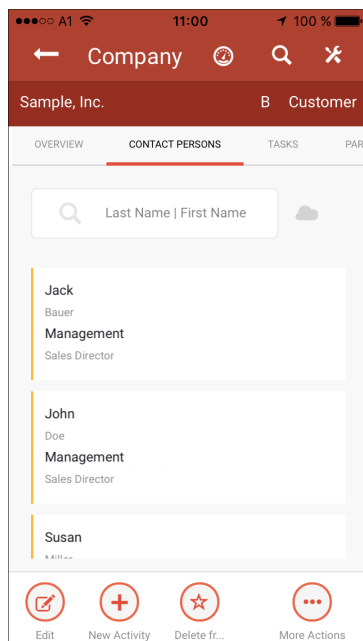


Lists are displayed in CRM.Client as shown below. Both lists use the (same) list control.

- A list displaying search results:



- A list displaying child info area records in the parent info area's Details page (for example, all the contact persons of a Company):



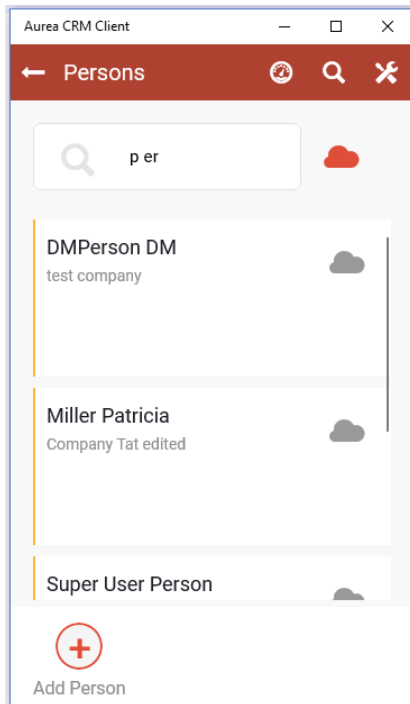
Notice the color coded bars on the left of the listed items in the screenshots above. The records can also display logos of their info area. The color coding and the logos for the info area are configured in the [Expand Configuration](#) on page 63.

The fields you selected are displayed in the same order as they are configured in the designer. You can combine the content of several fields with the `ColSpan` option, see [Combining Field Values](#) on page 53. You can also define whether a field is defined in bold or in a specific **Field Color**, see [Field Attributes](#) on page 50.

## Search Control

You can define the search fields available in the info area search and for record selectors in the Search control.

Only a text-based search field is available in CRM.Client. The search results are listed below the search field:



You can add all field types, but the following restrictions for adding fields apply:

- Currently fixed catalog data types are not supported. You can add them as filters.
- You can include computed fields ("Z fields") and catalog values in offline mode only, as Aurea CRM does not support wildcard text search in catalog values.
- If you have configured three fields of the current info area to be searched and the user enters more than one search criterion, then all the search criteria are searched in all the search fields, including the parent and related info areas.

## Field Attributes

The supported field attributes are described in this topic.

---

**Note:** The CRM.Client fields do not support the `Multi Line` attribute. So even when configured, it doesn't take an effect.

---

## ColSpan

Combines field values in view mode, see [Combining Field Values](#) on page 53.

## LoadCatalogs

Loads required catalogs without custom JavaScript module for an expand page.

## Extended Options

Learn about the extended option for field attributes.

You can use this argument to define specific options:

- `{"newLine" : true }` inserts a line break after the field.
- `{"supportsDecimals" : "false"}` defines that the user cannot enter decimals in this field
- `{"ShowFieldNameForTrueValue" : false }`: By default, Yes/No is displayed for boolean fields. You can define, that the field name is displayed instead, see [Format.ShowFieldNameForTrueValue](#). If you specify `{"ShowFieldNameForTrueValue" : false }` the `Format.ShowFieldNameForTrueValue` and the `Format.ShowFieldNameForTrueValue` Web Configuration parameters are ignored.
- `{"phone" : "tel"}` defines that you can call the telephone number in the field (if exists) directly from your device.

---

**Note:** For iOS devices, this option is only available with hardware that supports Apple Continuity. For more information see <https://www.apple.com/ios/whats-new/continuity/>.

---

The following extended options do not have default values:

## Field Color

Defines the color of the field value. You can also enter a hex value, for example, `#FF0000FF`, or in the ARGB format "`<red>;<green>;<blue>;<alpha>`", each entry to be given as a value between 0,0 and 1,0. The text values of *red*, *green*, *blue*, and *yellow* are also supported.

## Label Color (Details and MiniDetails only)

Defines the color of the field label. See [Field Color](#) for more information.

## No Label

Hides the field label.

---

**Note:** The field label is hidden by default in lists. Enabling this option results in the field label being displayed (which can be useful, for example, if you only display and sort numbers).

---

## No Multi Line

All line breaks and multiple spaces are removed.

---

**Note:** If you copy the text to the clipboard, the actual text (incl. line breaks and spaces) is copied.

---

## Render Hooks

You can influence how a field's content is displayed with the `Render Hook` field attribute. For example:

- `{"FieldType": "<type>"}`: Defines the number type. You only need to define this if the desired field type differs from the current field type.

Possible values: `F` (float) and `L` (long integer)

- `{"PercentField" :true }`: Formats the field as percentage field (the % character is appended) and also calculates its content as percentage value.

Example: 0,10 is displayed as 10 %

- `{"DecimalDigits": <number>}`: Defines how many decimal digits are displayed.

Example: `{"DecimalDigits":0}` does not display decimal digits

---

**Note:** You can also combine these formatting options, for example, to display a percentage field with one decimal point: `{"PercentField" :true , "DecimalDigits":1}`. These formatting options do not change the actual value in the database, but only influence how the value is displayed in CRM.Client.

---

- `{"GroupingSeparator":"false"}`: Defines that the digit grouping symbol is not displayed.

Possible values: `true` / `false`.

**Default:** `true`

Example: `{"GroupingSeparator":"false"}`

## Linked Fields

In many info areas so-called "Z" fields are available. These are write-protected/decoded fields whose values are entered automatically by the business logic, for example, system and link fields. These fields can cause problems in CRM.Client, therefore, always add these fields as linked fields and set a specific link, for example,:

Last Name (3) (PE#1) First Name (4) (PE#1)

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

For the `NewView` action call, linked fields are only supported if a record selector is defined. Example: The **Select Company** and **Select Person** fields in the `MA :New` Edit control.

## Combining Field Values

You can combine the content of more than one field in the Details and List control with the `ColSpan` field detail.

For example, you can combine the First Name and Last Name fields to be displayed as Name.

To combine several fields:

1. Add the fields to the desired control (Details, List).
2. Select the first field of the fields you want to combine and enter the number of fields you want to combine in the text field.

For example:

If you want to combine Street, Postal Code, City and Country to an address label, add these four fields to your control and set `ColSpan` to "1;2;1". In this case the field then consists of three lines: the first line contains the first field, the second one contains the next two fields and the third one contains the fourth field.



3. If you combine fields in View mode, the first field's **Label & Field Attributes** are applied. For example, if you combine four fields to an address label, the field's label is "Street", and if you combine two fields to a person's name, the label is "First Name". You can either hide the label with the **No Label** field detail, or you can define a field label.
4. You can also define a delimiter if more than one field is shown in one line. The following options are available:
  - "2:-" delimits with a minus.
  - "4 :b /b" delimits with the string " / " (each b is replaced by a blank allowing to trail blanks).
  - "3 :bcb" delimits with string " : " (each c is replaced by a colon).
  - "1 ; 2 ; 1:-" delimits the two strings in the second line with a hyphen.
  - "3 :FormatText:my:2" or 3:Text:my:1 combines three fields. The delimiter applied is a text from the `smartbook :my` text group. Wildcards (i. e. {1}, {2}, {3} or %1, %2, %3) can be applied and are replaced by the corresponding field values.

---

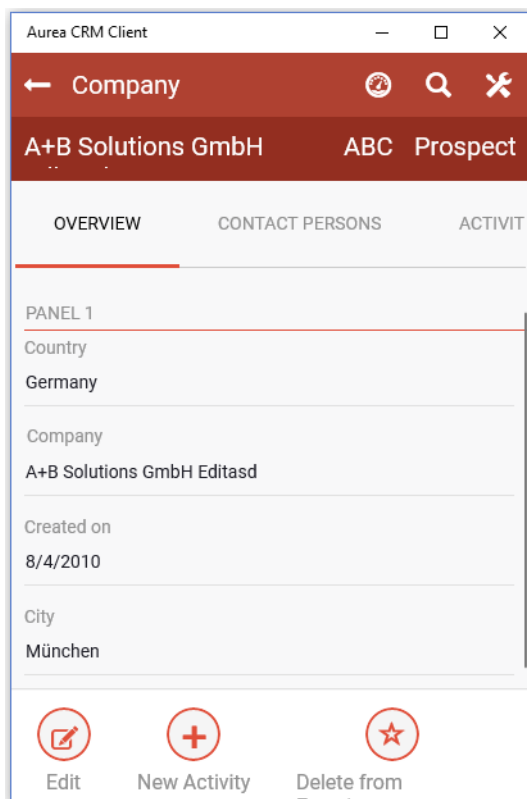
**Note:** Delimiters are only applied between two or more fields in the same line that contain values.

---

## Panels

You can add panels and then add fields to these panels.

In the following sample screenshot, a panel labeled "Panel 1" is configured to display the country, company name, city, and date the company was founded:



The size of each panel is adjusted automatically to its content.

Fields defined for Details and Edit controls are organized in panels.

In the **Panel Editor** you can define the following:

- **Label:** The text displayed as a subheader on the control.
- **Name:** The panel's name (the panel's ID).
- **Type:** You assign only GRID to your panel, see [Panel Types](#).
- **Add Fields to Panel:** If you want to add the fields of another existing panel, select the desired panel in the drop-down list. The fields of the selected panel are appended to this panel.
- **CRM Processes:** You can assign one or more CRM processes to the current panel.

For general information on how to handle panels, for example, on how to add or delete them, see [Configuring Panels in the Aurea CRM web Administrator Guide](#).

## Panel Types

You can assign the panel type to GRID, as for example:

---

**Note:** Another panel types are not supported by CRM.Client.

---

Use the **Panel Editor** of CRM.Designer to assign a type to a panel. Key in `GRID` in the **Type** field.

## Grid

By default, data is displayed in one column for each panel in the Details view.

You can define that data is displayed in two columns (= in a grid) called the Grid view. Empty fields are always displayed in a grid view.

You can also define a placeholder that you can add to the Grid view (for example, if you want that a specific field is always displayed as the first field in a line, you may want to define an empty placeholder). Add a field to the desired panel and set the `Empty` field attribute. This field is then displayed without label and without content.

The following sample screenshot shows the grid panel layout of the company details:

The screenshot shows the Aurea CRM Client interface. At the top, there's a header bar with the Aurea logo and navigation icons. Below the header, the main content area is divided into four tabs: OVERVIEW, CONTACT PERSONS, ACTIVITIES, and TASKS. The OVERVIEW tab is selected. Below the tabs, there's a grid panel layout. The grid has four columns. The first column is labeled 'City' and contains the value 'Created at'. The second column is labeled 'Created on' and contains the value '23:59'. The third column is labeled 'Country Code' and contains the value 'Viktor Vertrieb'. The fourth column is labeled '2017-12-31' and contains the value '+49'.

| SPEC-6448 Company |                 |                 |     |
|-------------------|-----------------|-----------------|-----|
| OVERVIEW          |                 | CONTACT PERSONS |     |
| PANEL             |                 |                 |     |
| City              | Created on      | 2017-12-31      |     |
| Created at        | 23:59           | Country Code    | +49 |
| Updated by        | Viktor Vertrieb |                 |     |

## Defining Sub-Lists (Related Data)

You can define that data from child info areas is also shown in Details or Edit view by adding additional panels to the Details or Edit control.

If you add an Edit control, the user can edit the child record's data as well.

To define sub-lists:

1. Add a panel.

The screenshot shows the Panel Editor interface. On the left, there's a list of panels: 'Base information 'Panel-1'', 'Person Details 'Panel-2'', and 'Activities 'Panel-3' (CHILDREN\_MA\_5)'. The 'Activities 'Panel-3' (CHILDREN\_MA\_5)' panel is selected. On the right, there's a 'Panel Editor' form with the following fields: Label (Activities), Name (Panel-3), Type (CHILDREN\_MA\_5), Add Fields to Panel (dropdown), CRM Processes (all), and a plus sign button to add more fields.

| Base information 'Panel-1' |                                              |
|----------------------------|----------------------------------------------|
| Form of address (0)        | First Name (3) Last Name (2) Department (11) |
| Company (2) (FI)           | FreeC1 (64)                                  |

| Person Details 'Panel-2' |                                      |
|--------------------------|--------------------------------------|
| Mobile Tel. (55)         | E-mail (9) Website (56) Country (19) |
| Postal Code (20)         | City (21) Street (22)                |

| Activities 'Panel-3' (CHILDREN_MA_5) |  |
|--------------------------------------|--|
|                                      |  |

| Panel Editor         |               |
|----------------------|---------------|
| Label:               | Activities    |
| Name:                | Panel-3       |
| Type:                | CHILDREN_MA_5 |
| Add Fields to Panel: |               |
| CRM Processes:       | all           |
| +                    |               |

2. Define a Label to be used as sub-list header. If undefined, the header text of the assigned Search&List configuration is applied.

## Search&List Configuration

The Search&List configuration defines the search fields, headers and search results displayed on the search page.

**SearchAndList 'FI'**

✓ Save ✗ Delete

Infoarea **Company (FI)**

Configuration Maria2\_pad

Field Group FI [Show FI](#)

Header Group FI [Show FI](#)

Context Menu <use default menu> -

Default Action <not selected> [Show SETTINGS SYNC](#)

Description Enter description here.

Help ID

Filter Name (UNKNOWN)

CRM.Client supports the following configurations in the Search&List configuration items:

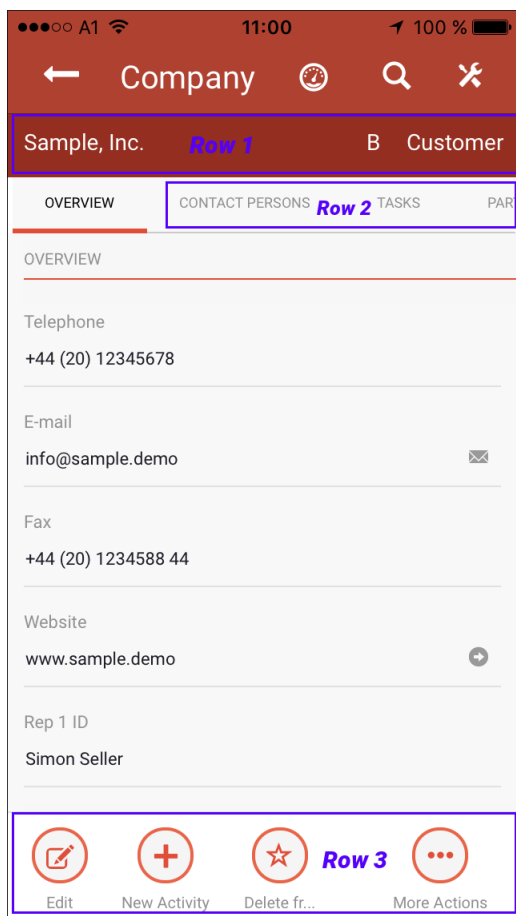
**Note:** The remaining items listed in the designer configuration are not supported.

- **Field Group:** Defines the layout of the search mask and list rows. The Search and List controls of the specified field group are used, >>[Search Control](#) and [List Control](#).
- **Header Group:** Defines the header group of the headers used on the page, see [Headers](#) on page 58.
- **Description:** Provide a description of this configuration. This description is available for view only within the Designer.
- **Filter Name:** The given filter limits the number of search results.

## Headers

Each page in CRM.Client contains a single header that can contain up to three rows. These rows display the following:

- **Row 1** displays the mini header which is a collapsed header without the expand. For more information, see [Defining a Header](#).
- **Row 2** displays the Info area tabs. For more information, see [Adding Info Areas \(Tabs\)](#).
- **Row 3** displays the Quick action buttons and **More Actions** menu. For more information, see [Adding Header Buttons](#).



## Header Group

You can define the contents of the headers displayed in CRM.Client, for example: Info area-specific headers are grouped in the **Header Groups** area of an info area.

| Header Groups |  |               |
|---------------|--|---------------|
| Name          |  | Configuration |
| FI            |  | update.tablet |
| SYSTEMINFO    |  | update.tablet |
| + Copy from   |  |               |

If no header group is defined, the corresponding header from the DEFAULT header group is applied. In the `update_tablet` configuration, the default header is the header with the info area ID as name, for example, `FI` is the default header group for the Company info area as shown in the screenshot above.

## Defining a Header

To define a header group:

1. Switch to the desired header group.
2. In the **Headers for Header Group [info area]** window, click on besides the desired view, for example, **Expand**.

The header group of the selected header (in this example the Expand header) is displayed.

Startpage > All Info Areas > Info Area FI > Header Group FI for Infoarea Company (FI)

### 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](#)
- [AddOffer](#)
- [UploadImage](#)
- [AddAppointment](#)

**Infoareas**

| Order | InfoAreaLink           | CRM Proc                     | Fieldgroup | Header   | Text    | Filter   | Tab Style | SubListParams | Menu | Search | Form Name | Flags |
|-------|------------------------|------------------------------|------------|----------|---------|----------|-----------|---------------|------|--------|-----------|-------|
|       | Person in Company (KP) | <input type="checkbox"/> all | no value   | no value | Default | no value |           |               |      | Sub_KP |           | 0     |
|       | Activity (MA)          | <input type="checkbox"/> all | no value   | no value | Default | no value |           |               |      | MA     |           | 0     |
|       | Offer (PR)             | <input type="checkbox"/> all | no value   | no value | Default | no value |           |               |      | PR     |           | 0     |
|       | Rep (ID)               |                              |            |          |         |          |           |               |      |        |           |       |

**Links**

Text Msg. Flags

Add a Link

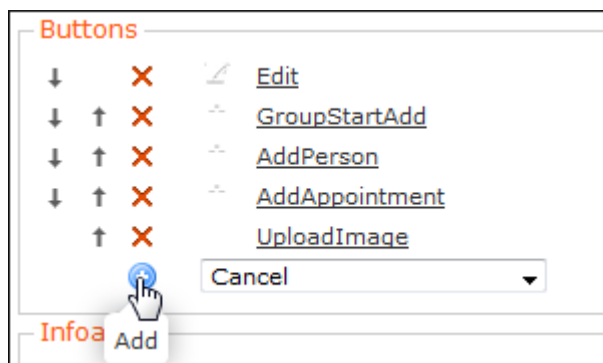
Only the following items are used in CRM.Client

- In the upper area you can define the **Text**, the text displayed in the header, whether buttons are displayed with text, what format is used, etc.
- In the lower area, you can define the following:
  - **Buttons**, see [Adding Header Buttons](#).
  - **Infoareas**, see [Adding Info Areas \(Tabs\)](#).
  - **Links**: Links are not supported in this release of CRM.Client.

## Adding Header Buttons

To define which functions are available in the application add the desired buttons:

1. Select a button from the drop-down list (the following example shows the FI header):



2. Click on  (**Add**).
3. You can click on a button to change its settings, see [Buttons](#) on page 30.
4. You can change the order using the  and  buttons.

---

**Note:** If you add more than three buttons, they are listed in the **More Actions** menu.

---

## Adding Info Areas (Tabs)

You can define that related data is displayed in a separate tab in CRM.Client by adding the desired info area in the header's **Infoareas** area.

---

**Note:** You can only use info areas with a data model link to the info area to which the header definition applies.

You cannot add a tab for info area-independent headers.

---

To add an info area (tab):

1. Select a button from the drop-down list (the following example shows the FI header):

| Order | InfoArea               | Link | CRM Proc | Fieldgroup | Header  | Text     | Filter   | Tab Style | SubListParams | Menu | Search | Form Name | Flags |
|-------|------------------------|------|----------|------------|---------|----------|----------|-----------|---------------|------|--------|-----------|-------|
| ↓     | Person in Company (KP) | all  | no value | no value   | Default | no value | no value |           |               | Goto | Sub_KP |           | 0     |
| ↓     | Activity (MA)          | all  | no value | no value   | Default | no value | no value |           |               | Goto | MA     |           | 0     |
| ↑     | Offer (PR)             | all  | no value | no value   | Default | no value | no value |           |               | Goto | PR     |           | 0     |
|       | Opportunity (Y1)       |      |          |            |         |          |          |           |               |      |        |           |       |

Links Add Text Msg. Flags Add a Link

- Click on **(Add)**.
  - You can change the order using the **↑** and **↓** buttons.
- The related data is displayed in CRM.Client as follows:

← Company

ABC Customer

A & S AG

OVERVIEW

CONTACT F

ACTIVITIES

TASKS

PARTICIPAT

LEADS

SPEC-6452 PANEL 1

Country

Germany

SPEC-6452 Company

A & S AG

Created on

8/4/2010

Edit

New Activity

Add to Favorites

The user can tap on the desired tab to display the data.

## Table Captions

You can use table captions to specify the field(s) of a record whose data is displayed in its header. For example, first and last name for a person or the company name.

To define a header label with a Table Caption:

- You can:
  - Switch to the desired info area and click on `Table Captions`.
  - Click on `Table Captions` (**Views** area) on the CRM.designer start page

Example:

**Table Caption 'KP'**

✓ Save ✗ Delete

Image <No Image> +

Prefix-Text {1} {2} ?

Format-Text

Description Enter description here.

Configuration Maria\_Test

|    | Info Area                | Link ID                  | Field            |
|----|--------------------------|--------------------------|------------------|
| 1. | ✗ Person in Company (KP) | <input type="checkbox"/> | First Name (3) ▼ |
| 2. | ✗ Person in Company (KP) | <input type="checkbox"/> | Last Name (2) ▼  |
|    | + Person in Company (KP) | <input type="checkbox"/> |                  |

2. You can define the following:

- **Format-Text:** You can define which fields added under **Field** are displayed in which order using placeholders. For example, in the screenshot above, First Name (1.) and Last Name (2.) were added. {1} {2} displays John Doe, {2}, {1} displays Doe, John.

If you do not define a **Format-Text**, no Table caption is displayed.

---

**Note:** If you want to display **Format-Text** without formatting (for example, display numbers without digit grouping symbol), enter the placeholder as follows {r1}

---

- **Description:** Provide a description of the configuration applied.

---

**Note:** The given description of the configuration is only available for view within CRM.designer.

---

3. In the lower area you can add the **Fields** you want to display in the table caption. The fields are numbered automatically. Use these numbers in the placeholders in the Format-Text field, for example, {1} {2} etc.

The following screenshot displays the infoarea name Company in the header label. See the screenshot below:

The screenshot shows a mobile application interface for a company record. At the top, there's a status bar with signal strength, 'A1', time '11:00', and battery '100 %'. Below is a red header bar with a back arrow, the title 'Company', and icons for a clock, search, and edit. The main content area has a red bar with 'Sample, Inc.' and 'B Customer'. Below this is a tabbed interface with 'OVERVIEW', 'CONTACT PERSONS', 'TASKS', and 'PARTS'. The 'OVERVIEW' tab is active, showing fields for Telephone (+44 (20) 12345678), E-mail (info@sample.demo), and Fax (+44 (20) 1234588 44).

## Expand Configuration

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

The screenshot shows the 'ExpandView Configuration' dialog for 'FI - Company'. It includes fields for Field Group, Header Group, Table Caption, Menu, Image, Color, Info Area Default Action, and Description. Below these are sections for View Parent Relations and Breadcrumb-Parent. A 'Recent Changes' table at the bottom shows a log of updates.

| When                | Operation | Who     | Details                          |
|---------------------|-----------|---------|----------------------------------|
| 2018-07-22 14:35:17 | Copy      | Default | From update.tablet to TestTablet |

The logo and the color coding for an info area record displayed in a list view and details view is configured using the **Image** and **Color** fields in the expand view as shown above.

The following options are supported:

- **Field Group:** Defines which Details control is used. For more information, see [Field Groups](#) on page 47.
- **Header Group:** Defines which `Expand` and `Expand.QuickActions` header is used.
- **Info Area Default Action:** Defines the default action that is executed when a user taps on a link.

## Virtual Info Areas

Virtual info areas are defined in Aurea CRM web.

Therefore, see the Aurea CRM web Administrator Guide on detailed information on how to configure and use them.

Only the function to display different data based on virtual info areas for one info area is available in CRM.Client.

Example:

You can define three different virtual info areas for the FI info area called VFI1, VFI2, and VFI3. You can assign different Details pages to each virtual info area. You can then display these three Details pages within the same FI list.

To use virtual info areas:

1. Create a `RecordView` action and assign `<ID>[Identifier]`:  
`<ID>` is replaced with the record's info area ID (FI, VFI1, etc., `[Identifier]` can be replaced with a string that determines the `Expand` configuration you want use for this feature.
2. Create a new context menu action and assign the `RecordView` page call. Set the `ConfigName` to `##[Identifier]`.
3. Create a separate `Expand` configuration for each info area named for example, `FI[Identifier]`, `VFI1[Identifier]`, `VFI2[Identifier]`, and `VFI3[Identifier]`, and assign the desired field group, header group, image etc. to each `Expand` configuration.  
  
If you do not define `Expand` configurations for the virtual info areas, the `Expand` configuration from the physical info area is applied.
4. Assign the previously created context menu action as default action to the corresponding `Search&List` configuration.

## Record Selectors

You can copy values from other records using a RecordSelector.

Two action templates are available for this purpose:

- `RecordSelect` for record selectors of type record, see [RecordSelect](#) on page 86.

### Record Selector Type Record

This record selector type copies field values from any info area to the current one. The user can search for a specific record and copy values by selecting the record. You can also use this function to link records.

The following example shows how to define that different persons or companies are linked to relationships:

1. Define a List control for your source field group in FI and KP. The source field group is the field group from which the values are copied from and defines the fields that are shown in the Record Selector's result list. If you also want to copy fields that are not be displayed in the list (e.g. FI-StaNo, FI-SerNo, KP-StaNo, KP-SerNo.), enable **Hide Field** in the **Field Attribute** area.
  - For the FI info area:
    1. Define a List control, in the following example FI:Select.

The screenshot shows the configuration interface for a 'Field-Group' named 'FI:Select' using a 'List' control. On the left, a search bar is present above a list of available fields including '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', and '15/C - Telephone'. The main area is divided into 'Fields' and 'Sort Fields' sections. The 'Fields' section contains a list with 'Company (2)', 'Synonym (3)', 'Country (5)', 'FI-StaNo (0)', and 'FI-SerNo (1)'. The 'Sort Fields' section has a placeholder 'Drop sort fields here...'. On the right, the 'Field Attributes' for 'Field 'FI-StaNo'' are shown, including 'Field Nr. 0, Type S', 'Label: <No Text>', 'Tool Tip: <No Text>', and 'Function: CoGr'. A 'Field Attributes' panel at the bottom right lists various options, with 'Hide Field' checked.

2. Enter the following functions in the Function field of the following fields:

| Field    | Function |
|----------|----------|
| Company  | Company  |
| FI-StaNo | CoGr     |
| FI-SerNo | CoNo     |

3. In the Search control of the same field group you can define the fields that are searched in the database (i.e the fields displayed in "ghost text" in CRM.Client's search field).

4. Add a Search&List configuration for your source copy field group:

The name of this Search&List configuration is used in the record selector's definition.

- Define the same items for the KP info area:
  1. Define a List control, in the following example KP:Select.

**Field-Group 'KP:Select', Control 'List'**

✓ Save ✗ Delete

Search Fieldname or -number 🔍

**0/X - Form of address**

**1/K - Acad. Title**

**2/C - Last Name**

**3/C - First Name**

4/C - Internal Address

5/C - Internal Name

6/C - Tel 1

7/C - Tel 2

8/C - Fax

9/C - E-mail 1

10/C - E-mail 2

11/K - Department

12/K - Position

13/K - Function

14/L - Rep ID

**Fields**

Form of address (0) Acad. Title (1)

Last Name (2) First Name (3)

FI\_StaNo (45) FI\_SerNo (46)

**PE\_StaNo (47)** PE\_SerNo (48)

**Sort Fields**

Drop sort fields here...

**Field "PE\_StaNo"**

Field Nr. 47, Type S

Label: <No Text> ✎

Tool Tip: <No Text> ✎

Function: PeGr

**Field Attributes**

☐ Add Rep Signature    ☐ All Columns  
☐ Auto-Wildcard Search    ☐ Bold  
☐ Bold Label    ☐ Browse for filename  
☐ Checkbox    ☐ Combo box  
☐ Document    ☐ Don't save  
☐ Don't save offline    ☐ Hide Catalog Button  
☒ Hide Field    ☐ Hyperlink

2. Enter the following functions in the Function field of the following fields:

| Field     | Function |
|-----------|----------|
| Last Name | LastName |
| FI-StaNo  | CoGr     |
| FI-SerNo  | CoNo     |
| KP-StaNo  | PeGr     |
| KP-SerNo  | PeNo     |

3. Search&List configuration:

2. Define the Edit control of the target field group. The user can call the record selector from this control.

In this example, the target field group is the PB field group of the PB info area:

3. Enter the following functions in the **Function** field to define the field values that are copied from the source field groups. Use the following syntax:

Related:<FunctionName>

Replace <FunctionName> with the **Function** ame you entered in the corresponding source field group.

| Field      | Function        |
|------------|-----------------|
| Company    | Related:Company |
| FI_1_StaNo | Related:CoGr    |

| Field      | Function         |
|------------|------------------|
| FI_1_SerNo | Related:CoNr     |
| Last Name  | Related:LastName |
| PE_1_StaNo | Related:PeGr     |
| PE_1_SerNo | Related:PeNo     |

4. Define the record selector by adding two context menu actions (one calling the record selector for FI, the second one for KP) and assign the `RecordSelect` action, see [RecordSelect](#) on page 86.

- Context menu action for companies (FI):

**Action for Menu FISelector**

Action Template: [RecordSelect](#)

virtual page for RecordSelect configuration

☒ Pass arguments as an object ?

**Input Arguments**

|              |                                                                     |
|--------------|---------------------------------------------------------------------|
| ConfigName   | FI:Select                                                           |
| LinkRecord   | FI                                                                  |
| ClearValues  | {'Related:Company','Related:FiGr','Related:FiNo','Related:LastN ... |
| TargetPrefix | Related:                                                            |
| LinkId       | 1                                                                   |

+ New Argumentname

**Output Arguments**

+ New Argumentname

Apply Cancel

- Context menu action for persons (KP):

**Action for Menu KPSelector**

Action Template: RecordSelect  
virtual page for RecordSelect configuration  
☐ Pass arguments as an object ?

**Input Arguments**

|              |           |
|--------------|-----------|
| ConfigName   | KP:Select |
| LinkRecord   | KP        |
| TargetPrefix | Related:  |
| LinkId       | 1         |

+ New Argumentname

**Output Arguments**

+ New Argumentname

Apply Cancel

- Switch to the Edit control's field group (you previously created) of the target info area (in this example, PB). Enter the desired context menu action name in the desired field's the `Select-Function` field.

Startpage > All Info Areas > Info Area PB > Field Group PB

**Field-Group 'PB', Control 'Edit' \***

Save Delete

Search Fieldname or -number

**Field "Company"**  
Field Nr. 42, Type Z

Label: <No Text>  
Tool Tip: <No Text>  
Function:

Modes: ☒ Details (All) ☐ New ☐ Update

☐ Image:  
☐ Item-Attributes:  
☐ Label-Color:  
☐ Label-Colspan:  
☐ Label-Field-Style:  
☐ Large Image Template:  
☐ Mask Configuration Option:  
☐ Multi Line:  
☐ Rep Filter:  
☐ Rowspan:  
☐ Small Image Template:  
☐ The zoom levels for which the field should be displayed.:  
☐ Delete Link:  
☐ Enhanced Search Field:  
☐ Hook Function:  
☐ Render Hook:  
☒ Select-Function:  
FISelector  
☐ Widget:

## Filters

Learn about filters info areas in CRM.client.

You can add filters to

- filter data in queries (client and server).
- define the offline data set (server only).

While offline data set filters are only executed on the server all other filter types must also be executable on the client.

Sometimes, custom filters defined for some records can prevent the CRM.Client application from synchronizing these records with the CRM.Client server, when these records are linked from other infoareas. To ensure that such linked records are synchronized with the CRM.Client device inspite of the preventive filters defined, set the `Sync.IncludeLinkedData` to `true`. For more information, see [Sync Parameters](#) on page 93.

### Variable Expressions (Filter Tokens)

Filter tokens (variables) are automatically replaced with the current value when the filter is applied. The following filter tokens are replaced by:

- `$curRep`: The `repId` of the current rep
- `$curTenant`: The current rep's tenant (not additional tenants!)
- `$curOrgGroup`, `$curDeputy`, `$curSuperior`: The current rep's organizational group, deputy and superior
- `$curGeoLongitude` or `$curGeoX`: The current longitude coordinate
- `$curGeoLatitude` or `$curGeoY`: The current latitude coordinate
- `$curDay`: The current date
- `$curfdWeek`: The date of Monday of the current week
- `$curfdMonth`: The first day of the current month
- `$curfdQuarter`: The first day of the current quarter
- `$curfdYear`: The first day of the current year
- `$curTimeSec`: The current time in the format `HHMMSS`
- `$curHour`: The current hour
- `$cur$$$Y`: The current year
- `?????####`: The current day, to be used in birthday queries, see [Birthday Filter](#).

You can add or subtract time periods like `+1d`, `-5d`, `+2w`, `-3m`, `+1y` for the following tokens: `$curDay`, `$curfdWeek`, `$curfdMonth`, `$curfdYear` and `?????####`.

You can use these expressions to hand over values to functions as variables with the `$par` prefix, for example:

- `$parDate:fdWeek`: the first week day of the date in `$parDate`
- `$parTime:Hour`: the hour of the time in `$parTime`

You can also add or subtract time periods, for example:

- `$parDate:Day+2d`: two days are added to the date in `$parDate`
- `$parDate:Hour+2h`: two hours are added to the hour of the date in `$parDate`

You can use the following expressions to hand over the current geographic coordinates to Template filters (for example, when you create a new company record on the client's premises, the device's current location is applied):

- `$curGeoX`: the current latitude coordinate
- `$curGeoY`: the current longitude coordinate

### Using Values from a Rep (ID) Record as Filter Tokens

You can also use values from ID records as filter tokens. For example, the `$parRep-Country` filter token is replaced by the rep's country.

To configure which tokens can be used:

1. Create a field group (for example, `ID.Copy`) and add the fields that you want to use as filter tokens to the List control.

Enter the desired function names into the **Function** field of each field, for example:

| Field           | Function   |
|-----------------|------------|
| Form of Address | RepGender  |
| Country         | RepCountry |
| Region          | RepRegion  |

2. Create a Search&List configuration with the same name as the field group and select the **Field Group** (in the example **ID.Copy**) and the **FilterName** (in the example **ID.curRep**). If this filter does not exist, the Filter Token always references the values of the first rep found in the ID table:

3. Enter the name of the Search&List configuration in the `RepCopySearchandLists` web configuration parameter, see [RepCopySearchAndLists](#):

You can apply these filter tokens in any query or filter by entering the following syntax: `$par<FunctionName>`.

## Using Values from a Record as Filter Tokens

For `NewView/EditView` action templates and for the quick add you can also define values from a specific record as filter tokens.

To define which tokens can be used:

1. Switch to the List control in the desired field group and add the fields that you want to use as filter tokens. You can also add records from other info areas that are linked (1:1 or n:1) to the current record.
2. Enter the desired token names in the **Function** field of the desired fields.
3. Set the `CopySourceFieldGroupName` input argument of the `NewView` and/or `EditView` action calls to the previously defined field group, see [NewView](#) on page 82 and [EditView](#) on page 80.
4. You can use these filter tokens to copy values from the current record when you open a new page, e. g. in Template filters.

## Birthday Filter

If you want to define a filter that returns all birthdays within the next feww days, you cannot define a time period in one criterion but have to add every day as follows:

## Adding Filters to the Action Call

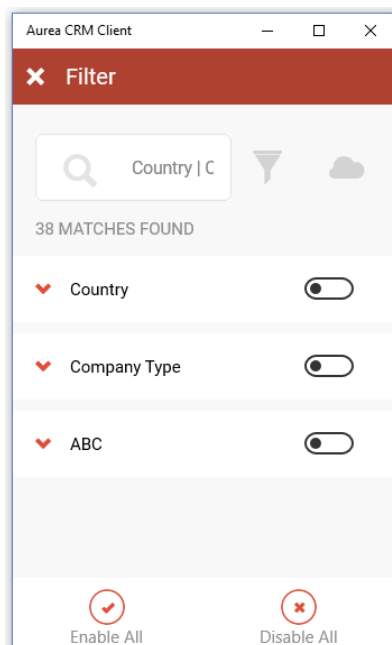
Up to six (depending on the action template) filters can be assigned to an action call.

If you need more filters, you can add them in the `AdditionalFilter` field (separated by semicolon).

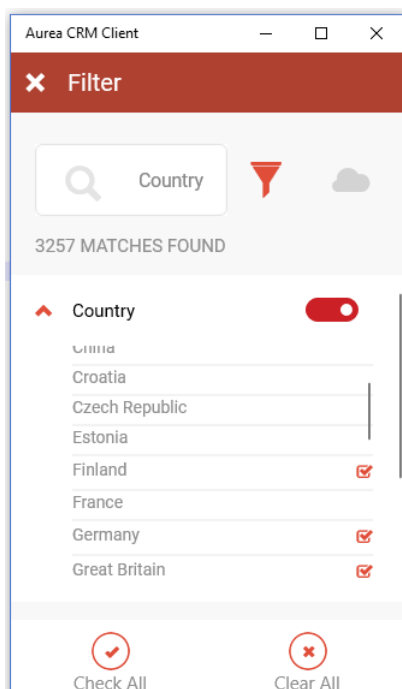
|                  |                       |   |   |
|------------------|-----------------------|---|---|
| Filter1          | MA.MyActivitiesFromTo | ▼ | ✎ |
| Filter2          | <No Value>            | ▼ | ✎ |
| Filter3          | <No Value>            | ▼ | ✎ |
| Filter4          | <No Value>            | ▼ | ✎ |
| Filter5          | <No Value>            | ▼ | ✎ |
| AdditionalFilter |                       |   |   |

A  (**Filter**) button is displayed in the application and the user can add and combine filters that are applied when searching the data.

If a filter has no parameter, it can be enabled or disabled (on/off switch as shown in the following screenshot).



If a filter has a parameter, the user must select one or more values from a list (check marks as shown in the following screenshot).



---

**Note:** If a selected filter is disabled, the selected values are still displayed but **not applied** anymore.

---

## Actions

Actions define what is "happening" in CRM.Client, for example, whether data is displayed in List or in Details view, whether they are in View or in Edit mode, and much more.

To assign actions, action templates are available. When assigning the actions to buttons or menu actions, you can specify input arguments that define which data is read, how it is read (from the server, locally), which filters are applied to limit the numbers of records, etc.

### Global Actions (not supported in this release)

Global actions are actions that are available in the control center's **Global** area, see [Control Center](#) on page 33.

Global actions are defined in the `$GlobalAction` context menu. This menu is included in the `update.tablet` configuration.

If this menu does not exist, no global actions are available.

By default, the `$GlobalAction` context menu contains the following context menu actions:

1. `INFO_SETTINGS`: Calls the `SettingsView` action, see [SettingsEditView and SettingsView](#).

## Action Templates

Action templates define what is displayed in the application, what actions the user can carry out, etc.

You can assign action templates to for example, buttons, context menu items etc. You can specify input arguments that define for example, the info areas called, the filters that are applied and more.

This topic only describes CRM.Client-specific action templates or CRM.Client-specific input arguments. For more information on action templates, see [Aurea CRM web Administrator Guide](#).

## General Input Arguments

The following input arguments apply to many action templates. The topics describing the action templates contain input arguments specific to them.

| Input Arguments                        | Description                                                                                                                                                                                                                                            |
|----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>AdditionalFilter</code>          | A comma separated list of additional filters that are applied.                                                                                                                                                                                         |
| <code>ConfigName</code>                | Name of the referenced Search&List configuration, fallback: field group of the same name.                                                                                                                                                              |
| <code>EnabledFilter</code>             | A comma separated list of filters that are enabled by default.                                                                                                                                                                                         |
| <code>Filter1 - Filter&lt;n&gt;</code> | Defines the filters that are available to the user when he taps on the  ( <b>Filter</b> ) button. The number of available filters depends on the action template. |
| <code>FilterName</code>                | The name of an additional fixed filter that is applied.                                                                                                                                                                                                |

| Input Arguments                                           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>hideOnlineOfflineButton</code>                      | <p>If set to <code>true</code>, the  (online/offline) button allowing the user to search data online is not displayed. Only the offline search is available.</p> <p>Default value: <code>false</code></p>                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>IgnoreSavedAction</code>                            | <p>If set to <code>true</code>, the action defined by the <code>SavedAction</code> argument is ignored and after adding a new child record, its root record is displayed.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>InfoArea</code>                                     | <p>Defines the info area for which records are displayed.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>RecordId</code>                                     | <p>Defines the record that is the parent or root record, usually the current record (for example, in edit mode, the record to be edited, and in New mode, the linked parent record, the record from which the New mode was started).</p> <p>If left empty, the parent record's <code>RecordId</code> is applied, for example, for samples the activity's <code>RecordId</code>.</p>                                                                                                                                                                                                                                                                                                                                      |
| <code>RequestOption</code> OR<br><code>RequestMode</code> | <p>Defines how data is read. Depending on the action called, not all options are applicable. The default value also depends on the action.</p> <p>The following options are available:</p> <ul style="list-style-type: none"> <li>• <code>Fastest</code>: Data is read offline, if available. If not found offline, the data is searched online if a server connection is available.</li> <li>• <code>Best</code>: Data is searched for online. If no network connection is available, data is read offline</li> <li>• <code>Online</code>: Data is always read online (data cannot be displayed if no network connection was established)</li> <li>• <code>Offline</code>: Data is always read offline only.</li> </ul> |

| Input Arguments            | Description                                                                                                                    |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| <code>SearchOptions</code> | Available option:<br><br><code>NoEmptySearch</code> : If the search field is empty, the search is not started.                 |
| <code>SortSequence</code>  | If set to <code>ASC</code> (default), data is sorted in ascending order. <code>DESC</code> sorts the data in descending order. |

## SigningConfig Input Argument

The `SigningConfig` input argument defines that an e-mail can only be sent after the report was signed by the customer, for example, an order.

**Default value:** `{}`

The following options can be entered in JSON syntax:

- `"sign":false`: Defines that no signature is required.

**Default:** `true`

- `"upload":false`: Defines that the client report is not uploaded to the database. The report is then only attached to the e-mail.

**Default:** `true`

- `emptySignatureName` : A comma-separated list of empty signature images. An empty signature image must be given for each required signature.

**Default:** `Button :PleaseSign .`

- `signatureImageTagName` : A comma-separated list of XML tags that references the signature images. A tag must be given for each required signature.

**Default:** `SignatureImage.`

- `signatureImageId`: Name of the `img` tag's `id` attribute in the XSL file.

**Default:** `img-signature`

- `lineWidth`: Font width

**Default:** `3`

- `strokeColor`: Font color

**Default:** `0;0;1;1` (= blue)

- `imageWidth`: Width of the image file containing the signature

**Default:** `300` (px)

- `signedReportFileName`: Name of the generated PDF file

**Default:** `SampleReport.pdf`

- `signedReportFileNameDateFormat`: Date format in case a date parameter is appended to the file name

**Default:** `yy-mm-dd`

- `signatureTitle` : Name of a field group. This field group's List control is applied and the field containing the `SIGNATURETITLES` function name is searched. This function may contain `{ $var }` placeholders that are replaced according to the given `CopySourceFieldGroupName` argument in the `ClientReport` action.

You can also define that more than one signature must be given. The following example requires two signatures:

```
{ "emptySignatureName": "Button:PleaseSign;Button:PleaseSign2",
  "signatureImageTagName": "SignatureImage;SignatureImage2", "sign": true,
  "upload": true }
```

## ConflictsView

This action template can be added to the System area to display all uncommitted requests.

The user can either export the requests or delete individual ones (for example, if they cause the system to crash when switching from offline to online mode).

## DashboardView

Learn about the Dashboard view display in Action template.

This action template displays the dashboard.

| Input Argument | Description                                                           |
|----------------|-----------------------------------------------------------------------|
| ConfigName     | Name of the generic form that is displayed.                           |
| ConfigTabNr    |                                                                       |
| HeaderName     | Defines the header applied. Syntax:<br><headergroupname>.<headername> |

## DataSync

This action template allows to call the Synchronization page from any other page in the application.

No input arguments are required.

## DebugView

This action template displays the Debug page.

No input arguments are required.

## EditView

This action template displays the Edit view that allows the user to edit an existing record's data.

| Input Argument          | Description                                                                                                                                                                                                                                                                                                                                                                   |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>ExpandName</code> | Name of the Expand configuration to be applied.                                                                                                                                                                                                                                                                                                                               |
| <code>RecordId</code>   | Defines the record that is the parent or root record, usually the current record (for example, in edit mode, the record to be edited, and in New mode, the linked parent record, the record from which the New mode was started).<br><br>If left empty, the parent record's <code>RecordId</code> is applied, for example, for samples the activity's <code>RecordId</code> . |

The following parameters are not supported in this release:

- `CopySourceFieldGroupName`
- `RequestMode`
- `SavedAction`
- `SyncParentInfoAreaIds`
- `RightsFilterName`
- `CopyRecordId`
- `RefreshFilterTokens`
- `Options`
- `ClientRightsFilterName`

## InsightBoardItem

This action template displays the Insight Board on Details pages.

Action for Menu Form:Details

Action Template: InsightBoardItem

☐ Pass arguments as an object ?

Input Arguments

- ! ConfigName: AP
- ContextMenu: TE\_CN\_InsB\_LIST\_MA
- RequestOption: Fastest
- ForceActionStyle: true
- ✖ BackgroundColor: 0xFFFF00
- + New Argumentname

Apply Cancel

| Input Argument   | Description                                                                                                                                                                                                                |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ConfigName       | The name of the default tile's Search&List configuration. Filters limiting the number of hits can be referenced in the Search&List configuration.                                                                          |
| ContextMenu      | Name of an alternative context menu action. If given, the default configuration is overridden.<br><br>Must be given for Insight Boards that are displayed on the login page, see <a href="#">Insight Board</a> on page 94. |
| RequestOption    | Defines how data is requested, see <code>RequestOption</code> argument in <a href="#">General Input Arguments</a> on page 76.<br><b>Default:</b> <code>Offline</code>                                                      |
| ForceActionStyle | Defines that instead of a number an icon is displayed on the tile.<br><br>Possible values: <code>true</code>   <code>false</code><br><b>Default:</b> <code>false</code>                                                    |
| BackgroundColor  | Allows you to provide a background color to the Insight Board item. For example, the value can be set to <code>0xFFFF00</code> .                                                                                           |

## NewOrEditView

This action template can be applied in specific cases: If a child records exists, the Edit mode is applied, if no record exists, the New mode is applied.

| Input Argument     | Description                                                                                                                       |
|--------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| Mode               | Takes the fixed value of EditOrNew mode. Other modes are not supported and if set to a different value, the set value is ignored. |
| ConfigName         | Name of the referenced View configuration.                                                                                        |
| LinkRecord         | The record linked to the current record.                                                                                          |
| NewOrEditViewQuery | Name of the query to be executed.                                                                                                 |

The following parameters are not supported:

- LinkRecordId - ID of the linked record.

## NewView

This action templates displays the New view allowing the user to enter data for a new record.

| Input Argument | Description                                                                                                                 |
|----------------|-----------------------------------------------------------------------------------------------------------------------------|
| Mode           | It takes a fixed value of <code>New</code> . Setting it to any other value does not change the fixed value, and is ignored. |
| ConfigName     | Name of the referenced view configuration.                                                                                  |

The following parameters are not supported in this release:

- AdditionalParameters
- LinkId
- LinkId2
- LinkRecordId2
- ParentLink
- RequestMode
- SavedAction
- SyncParentInfoAreaIds
- LinkRecordId

- TemplateFilterName
- CopySourceFieldGroupName

## OrganizerAction

In some cases, you can assign the `OrganizerAction` action template and define the `Action` input argument to determine which functions are carried out.

| Input Argument     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Action             | <p>Defines which functions are executed. The following actions are available:</p> <ul style="list-style-type: none"> <li>• <code>switchToEdit</code>: enables edit mode.</li> <li>• <code>deleteRecord</code>:</li> <li>• <code>RecordId</code></li> </ul> <p>The following actions are not available in this release of CRM.Client:</p> <ul style="list-style-type: none"> <li>• <code>Edit</code>: Enables the edit mode.</li> <li>• <code>Delete</code>: To delete existing records.</li> <li>• <code>switchToEditSerialEntry</code>: Enables the quick add's Edit mode.</li> <li>• <code>ModifyRecord</code>: To apply pre-defined changes to a record.</li> <li>• <code>RecordSwitch</code>: Defines the follow-up action to be executed.</li> </ul> <p>Example: A report was signed and cannot be changed anymore. If you send this report by e-mail, the report is generated anew (possibly containing different data then the signed one). You can prevent this behavior using the <code>RecordSwitch</code> action.</p> <ul style="list-style-type: none"> <li>• <code>toggleFavorite</code>: Adds/removes a record to the Favorite list.</li> </ul> |
| ContextMenu-Action | Name of a button that has the desired context menu action assigned.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| TemplateFilter     | Name of the filter that defines pre-defined values that are applied.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

The following parameters are not supported in this release:

- The following Actions are not supported in this release of CRM.Client:
  - `Edit`: Enables the edit mode.
  - `Delete`: To delete existing records.
  - `switchToEditSerialEntry`: Enables the quick add's Edit mode.

- `ModifyRecord`: To apply pre-defined changes to a record.
- `RecordSwitch`: Defines the follow-up action to be executed.

Example: A report was signed and cannot be changed anymore. If you send this report by e-mail, the report is generated anew (possibly containing different data then the signed one). You can prevent this behavior using the `RecordSwitch` action.

- `toggleFavorite`: Adds/removes a record to the Favorite list.
- `ContextMenuAction`
- `Option`
- `RequestMode`
- `RightsFilter`
- `TemplateFilter`

## Config Input Argument

Define the options for the photo upload in JSON syntax:

- `compression`: The JPEG compression rate. A float value between 0 (bad quality) and 1 (best quality).

**Default:** 0.8

- `fileNamePattern`: File name, use the `%1` placeholder for the `RecordId`.

**Default:** `photo.jpg`.

- `textPattern`: The options for the selected size. You can use the following placeholders:

- `%1`: Contains the file name
- `%2`: Width in pixel
- `%3`: Height in pixel
- `%4`: File size

**Default:** `%1 (%2 * %3, %4)`

- `sizes` (int array): Determines the image side that is larger. This value is used to scale the image. The user can select the desired size when uploading the picture.

**Default:** `[320, 640, 1280, 0]`: Maximum of four values possible:

- 320: the longer side is scaled to 320 pixel if larger
- 640: the longer side is scaled to 640 pixel if larger
- 1280: the longer side is scaled to 1280 pixel if larger
- 0: the current size is applied.

- `skipOne true|false`: Defines whether the selected size can be exceeded, if only one option is available.

**Default:** `false`.

- `imageNameEditable true | false`: Defines whether the user can enter in the image's name.

**Default:** `false`

- `startsWithCamera (as of version 1.5.2) true | false`: Defines whether the upload is started directly from the camera or from the photo roll.

**Default:** `false`

Example:

```
{"compression": 0.8, "sizes": [320 ,640,1280,0 ], "fileNamePattern": "photo.jpg", "skipOne": false, "textPattern": "%1 (%2 * %3, %4)", "imageNameEditable": false}
```

If you do not enter a JSON script, default values are applied.

## RecordListView

This action template displays records in List view.

| Input Argument | Description                                                  |
|----------------|--------------------------------------------------------------|
| Enabled Filter | Comma-separated list of filters that are enabled by default. |
|                | <b>Note:</b> Do not enter filters that contain parameters.   |
|                |                                                              |
| MaxResults     | Maximum number of records displayed in list.                 |

The following filters defined in [General Input Arguments](#) on page 76 are supported by RecordListView action template:

- AdditionalFilter
- hideOnlineOfflineButton
- InfoArea
- ConfigName
- Filter1 - Filter5

The following filters are not supported by RecordListView action template:

- ListStyle
- Sections
- SearchPageMode
- ParentLink
- SwipeDetailRecords
- LinkRecord
- LinkId
- FilterName
- FullTextSearch
- SearchOptions
- DetailsActionSwitchFilterName
- MapView
- CalendarView
- CalendarPopOverConfig

## RecordSelect

A record selector allows you to select data from another info area and "copy" them to the current info area.

You can use the `RecordSelect` action template for record selectors of type `record`.

---

**Note:** The `RecordSelect` action replaces the JSON script formerly used. Do not use the JSON script anymore.

---

| Input Argument       | Description                                                                                                                                                                     |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| LinkRecord           | The info area from which field values are copied.                                                                                                                               |
| TargetLinkInfoAreaId | Info area of the record that is linked with the record selector.<br>You can also specify <code>NoLink</code> , for example, if you use a record selector in several info areas. |
| TargetLinkId         | The <code>LinkId</code> set by the record selector.                                                                                                                             |
| ClearValues          | JSON array with function names to define the fields whose values are cleared before a new link is assigned.                                                                     |

| Input Argument      | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| FixedValues         | JSON dictionary containing fixed values to be applied when the record selector is applied. Syntax:<br><pre>{"FunctionName": "Value"}</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| TargetPrefix        | Prefix for the function names defined in the target configuration ( <code>ConfigName</code> ). This prefix is combined with the identifiers ( <b>Function</b> field) specified for the source copy fields in the target field group.                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| LinkId              | The given <code>LinkId</code> is used for the record selector's Search&List configuration.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| FormattedParameters | You can define certain format parameters (using JSON syntax).<br>Examples: <ul style="list-style-type: none"> <li>If the source List control of the record selector contains fields with the <code>FirstName</code> and <code>LastName</code> <b>Functions</b> assigned, you can enter <code>{"FullName": "%FirstName %LastName"}</code>. This allows you to assign a function called <code>FullName</code> in the desired target Edit control resulting in first and last name being copied to the target record.</li> <li>If you enter <code>{"FullName": "my :2 "}</code>, the text no. 2 from the <code>smarbook:my</code> textgroup is applied (a language-dependent string is displayed).</li> </ul> |
| HideStandardFilter  | Defines whether default filters (for example, recently used) are displayed.<br>Default value: <code>false</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| DisableLinkOption   | If set to <code>true</code> , a filter button allowing the user to show all records is displayed.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| IgnoreFieldInfo     | If set to <code>true</code> , the read-only flag is ignored.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

## RecordView

This action template displays the details of a record in the Details view.

The following parameters are not supported in this release:

- RequestOption - takes a fixed value: **Best**.

---

**Note:** By default CRM.Client uses the best available source to retrieve a record's data. Data is read from the offline cache, if no network connection is available. If a network connection is available then data is retrieved from the online database.

---

- OrganizerRefresh

---

**Note:** The details organizer is always refreshed when a record's data is requested anew.

---

- SyncParentInfoAreaId

---

**Note:** By default CRM.Client always synchronizes a record's parent infoarea.

---

## RecordViewSpecific

This action template displays a record that has been specified by the `RecordId`.

| Input Argument | Description                            |
|----------------|----------------------------------------|
| ConfigName     | The name of the configuration applied. |
| RecordId       | Id of the record.                      |

## SystemInfoPage

The `SystemInfoPage` action template displays the system information in the control center.

| Input Argument | Description                                                                                                                                                                                                                                                                                 |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| fieldgroup     | <p>Defines the field group that is used to display the contents of the system information.</p> <p>The <code>SYSTEMINFO</code> field group in the Company (FI) info area exists in the <code>update.tablet</code> configuration. However, you can assign a field group of any info area.</p> |

## WebView

This action template is used to display web pages in CRM Client.

| Input Argument | Description                                                                             |
|----------------|-----------------------------------------------------------------------------------------|
| HeaderName     | Name of the special header that is applied if the web page is displayed in an own page. |
| Height         | The height of the panel if the web page is displayed in the Details view.               |
| ReportType     | <code>Url</code> (fixed value).                                                         |
| Url            | The web pages URL. Use <code>index</code> if you want to display an internal tutorial.  |

## Web Configuration Parameters and Layouts

Web Configuration parameters influence CRM.Client.

Some of the parameters described in this topic are not included in the default configuration. If needed, you can add them.

Information on how to set parameter values or add new parameters you find in the Web Configuration topic in the Aurea CRM web Administrator Guide.

## Client Timeout Parameters

Learn about the client timeout parameter for CRM.client.

### ClientRequestTimeout

This parameter is ignored under iOS6. Requests are aborted after 60 seconds by the client.

This timeout is often set to several minutes to ensure that for example, the full synchronization request can be carried out without encountering a timeout. This timeout is also the time span until a "Server not available" error is generated if a request is not responded by the server.

**Type:** Character

**Default client timeout:** 60 seconds

### ClientSessionTimeout

Defines the time span in minutes after which the user has to log on anew. This can occur if the user works with other applications and CRM.Client is not on the screen or the device switches off automatically. However, the operating system can also switch off the application (this can happen before the timeout set in this parameter occurs).

**Type:** Character

## Format Parameters

Learn about the format parameter for CRM.client.

### Format.ShowFieldNameForTrueValue

If enabled, the field label is displayed for Boolean fields (instead of YES/NO).

No value is available for FALSE (NO). If you generate a client report (i. e. an xml export), the whole xml element is omitted. To export the xml element, set the `Format.ShowYesNoInReports` parameter.

**Type:** Checkbox

## Log Parameters

Learn about the log parameter for CRM.client.

---

**Note:** To override these parameters, see [Overruling Default Log Settings in the CRM.Client](#).

---

### Log.Level

Defines which occurrences are written to the log file. Takes the log levels values from 0 to 3, where: 0-None; 1-Error (Default); 2-Warning; 3-Info.

**Type:** Number

### **Log.Class**

Name of the class to be logged. For development purposes only.

**Type:** Character

### **Log.Config**

If enabled, configuration changes are logged.

**Type:** Checkbox

### **Log.Method**

Name of the class to be logged. For development purposes only.

**Type:** Character

### **Log.Network**

If enabled, network activities are logged.

**Type:** Checkbox

### **Log.Requests**

If enabled, all requests are logged.

**Type:** Checkbox

### **Log.Result**

If enabled, each result row is logged is logged.

**Type:** Checkbox

### **Log.Statements**

If enabled, SQL log changes are logged.

**Type:** Checkbox

### **Log.UpSync**

If enabled, all UpSync activities are logged.

**Type:** Checkbox

### **LogReport.EmailAddress**

Defines the e-mail address of the person that receives the client log files.

## Login Parameters

Learn about the Login parameter for CRM.client.

### Login.InsightBoard

You can enter the name of an existing Insight Board configuration (the context menu name). This Insight Board must have the `contextMenu` argument given, see [Insight Board](#) on page 94.

### Login.SavePasswordAllowed

Defines that the password is stored on the device for future comparison.

This parameter must be set if you want to define that users can log on automatically to the application, see [Including User Credentials in the Registration](#) on page 13.

## Search Parameters

Learn about the Search parameter for CRM.client.

### Search.AutoSwitchToOffline

If enabled, any search criteria applied in online mode are retained and applied in the offline mode. By default, search criteria applied in online mode are discarded when the user switches to the offline mode.

Example: A user switches to the online search mode (using the "Cloud" button) and then changes his search criteria. If this parameter is set, the new search criteria are also applied to an offline search.

---

**Note:** Find-as-you-type is only available in the offline mode.

---

**Type:** Checkbox

### Search.OfflineDelayTime

Defines the delay time for the find-as-you-type function. If the user does not edit the search field's content, after 0.3 seconds the search is started. Increase the value of this parameter if you encounter performance problems ("slow searches").

**Type:** Character

**Default value:** 0.3 (seconds)

### Search.OnlineDelayTime

Defines the delay time for the find-as-you-type function. If the user does not edit the search field's content, after 0.3 seconds the search is started. ("slow searches")

**Type:** Character

**Default value:** 0.3 (seconds)

## Sync Parameters

Learn about the Sync parameter for CRM.client.

### **Sync.FullSyncBlockIntervalDays**

Number of days after the last full synchronization when the user cannot access the application anymore.

This parameter and the `Sync.FullSyncIntervalDays` parameter are mutually exclusive.

**Type:** Character

**Default value:** 28 (days after the last full synchronization)

### **Sync.FullSyncIntervalDays**

Number of days after the last full synchronization when a full synchronization is carried out automatically.

This parameter and the `Sync.FullSyncBlockIntervalDays` parameter are mutually exclusive.

**Type:** Character

**Default value:** 21 (days after the last full synchronization)

### **Sync.FullSyncDate**

Forces all users who login after the specified date and time to carry out a full synchronization of their CRM.Client application.

**Type:** DateTime

### **Sync.FullSyncWarnIntervalDays**

Number of days after the last full synchronization when a warning is displayed.

**Type:** Character

**Default value:** 7 (days after the last full synchronization)

## Other Parameters

Learn about the other parameter for CRM.client.

### **Delete.Ask**

If enabled, a confirmation dialog box is displayed, when the user deletes a record.

**Default value:** `false`

**Type:** Checkbox

## StartPage

Defines the page displayed after the user logs in or after he taps on **Reset All Data and Reinitialize**.

## ErrorHandling.ToasterTimeout

Specify the duration in seconds, for which a Toast displays an error message on screen, before it auto-closes. The default value is 10 seconds.

---

**Note:** This timeout value applied only to messages a Toast displays when you are logged-in to the CRM.Client.

---

**Type:** Number

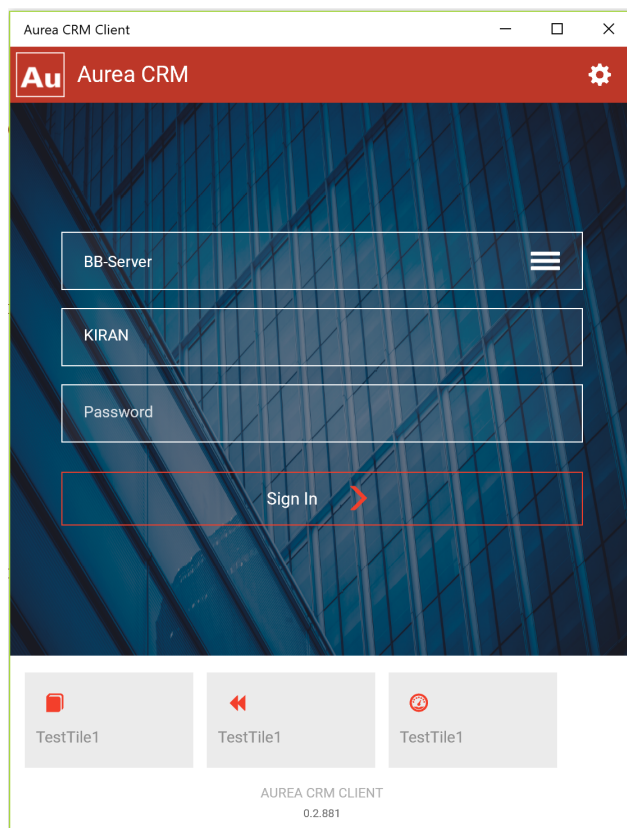
**Default Value:** 10 seconds

# Insight Board

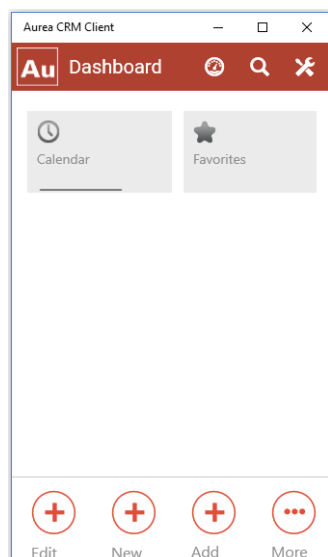
Learn about the insight boards available for CRM.Client.

Three types of Insight Board are available:

- Login Screen: You can add an Insight Board to the login screen, For more information, see [Adding an Insight Board to the Login Page](#) on page 29.



- **Start Page:** Supports tiles that is an InsightBoardItem configured with the record list view to display the info areas supported. For more information, see [Adding the Insight Board to the Start Page](#).



- **Custom Insight Boards:** You can define your own Insight Boards and add them to the start page.

Insight Boards can contain any number of tiles. If a user taps on a tile, the assigned action (usually displaying a list) is executed.

For more information on configuring an insight board, see [Configuring an Insight Board](#):

## Configuring an Insight Board

To configure an Insight Board:

1. Switch to the **Context Menu** area and add a new context menu, for example, `InsightBoard`.
2. Add a context menu action for each tile and assign the `InsightBoardItem`, see [InsightBoardItem](#) on page 80.
  - You must at least specify the `ConfigName`, that is, the name of a Search&List configuration. This results in a tile displaying the number of records and switching to the record list when the user taps the tile.
  - You can provide a background color for an Insight Board item by specifying the hexcode of the required color in the `BackgroundColor` input argument.
  - If you want to call any action template **except** `RecordListView`, you need to also specify the `ContextMenu` input argument. The action assigned to the given context menu action is executed, when the user taps on the tile.
3. Specify the `InsightBoard.TitleEnabled` Web Configuration parameter if you want to display a header, see [InsightBoard.TitleEnabled](#).

## Adding the Insight Board to the Start Page

To add an Insight Board to the start page:

1. Switch to the **Legacy Forms** area and open the **StartPage** form.
2. Add a row of type `InsightBoard` and enter the following parameters (separated by semicolon) in the **Func** field.
  - `MenuName`: Name of the Insight Board's context menu.
  - `RecordId`: for example, `ID.$curRep` for the current rep.
  - `Param1`: Name (with the \$ prefix) of a dependent start page component. For example, the default `StartPage` Legacy Form contains a `DatePicker` row with `Calendar` as **Valuename** assigned. You can therefore use `$Calendar` to display a tile with the selected day.

Example:

```
InsightBoard;ID.$curRep;$Calendar
```

3. You can define further options in JSON syntax in the **Options** field:
  - `MaxVisibleItems`: Limits the number of tiles displayed. If more tiles exist, the user can slide within the Insight Board.

Example:

```
{"MaxVisibleItems":"2","Vertical":"1"}
```

# 3

## Troubleshooting

---

### CRM.client troubleshooting

In this topic you find information on how to solve known issues.

### General Checks

General check for troubleshooting.

If problems occur, check the following:

- Check the configuration settings in the `web.config` file.
- If an error occurs, always check the `update_net.log` of the Aurea CRM web installation.
- Check whether .net 4.0 is installed and used for the application pool.
- Check whether the **ISAPI and CGI Restrictions** for the application pool's .net Framework 4.0 are set to `allowed`.

### Problems Concerning Rights

Learn to troubleshoot rights issues.

If right problems occur:

- Check if the Mobile Services right was assigned to the user in the login configuration (`mmcfg.exe`).
- Check if single sign-on was assigned to the user.
- If messages in the `update_net.log` state that an info area or a field does not exist, usually the field is missing in the CRM.designer data model. Try to upload the data model to the CRM.designer database using Aurea CRM web and carry out your last action again. You must also update CRM.Client's data model, see [Data Model](#) on page 20.

## Connection Problems

Learn how to troubleshoot connection problems.

If server connections problems occur, check the following:

- Check in the iOS or Windows 10 **Settings** whether you are connected to the internet or to corporate WLAN. You can also try to open a web site (e. g. [www.aurea.com](http://www.aurea.com)) with a browser (for example, Safari).
- If you can access a web site, check if you can access Aurea CRM web with the browser by entering the URL. On iOS devices, if an error message is displayed stating that Aurea CRM web does not support Safari, you were able to connect to the server.

## Login Errors

Learn how to troubleshoot login errors.

The following errors may occur during login:

- If a user logs on the first time and "Hello WWW" is displayed, the server registration link was probably wrong.

A server can be de-registered with:

```
crmpad://removeServer?name=<Servername>
```

- If the initial login failed (error message: "Login failed – No data has been synchronized for this user yet. Please check internet connection & user name" or similar), you can check whether authentication was successful by entering one of the following URLs into Safari:

- **For anonymous access (with username/password):** `<myServer>/mobile.axd?Service=Authenticate&Username=<name>&Password=<password>&ClientVersion=<AppStoreVersionNumber>`

AppStore version number: for example, 1.1

- **For authenticated access (with sso):**
- `<myServer>/mobile.axd?Service=Authenticate&ClientVersion=<AppStoreVersionNumber>`

AppStore version number: for example, 1.1

The authentication was successful if content similar to the following is displayed:

```
{
  "authenticated": true,
  "sessioninfo": [
    "200",
    [
      ["repName", "Martin Praunias"],
      ["repId", "010001909"],
      ["repGroupId", "000000000"],
      ["repDeputyId", ""],
      ["repSuperiorId", ""],
      ["repGroupId", ""],
      ["tenantNo", "0"],
      ["tenantName", ""],
      ["tenantAdd", ""],
      ["datamodelVersion", "1.0"],
      ["clientRequestTimeout", ""]
    ]
  ]
}
```

- If the login failed after the user has already logged in successfully at least once (error message: "Login Failed – Incorrect user name or password. Please check our login data and try again" or similar), try the following:

1. Check the client log file for a response from the server with an error.
2. Check the server log file for the error message.
3. If you could not identify the cause, try the links for the initial login fail find out if a correct response from the server is returned. If yes, try the following statement to find out if the data model or the configuration is corrupted:

```
CRM.server/mobile.axd?Service=Synchronization&SyncDataModel=true&
SyncConfiguration=true&SyncCatalogs=true
```

## Performance Problems

Learn how to troubleshoot performance issue.

### CRM.Client Server Request Timeout

Each CRM.Client server request is subject to two timeouts. The lower value determines the maximum execution time:

1. The server's ASP.NET timeout after which the server request is canceled (default = 90 seconds). You can increase this value in the `web.Config` file: `<!-- Increase script timeout to five minutes --> <httpRuntime executionTimeout="300" ... other configuration attributes .../>`
2. The default client request timeout of the CRM.Client is 60 seconds. You can increase this value with the `ClientRequestTimeout` Web Configuration parameter, see [Web Configuration Parameters and Layouts](#) on page 89.

---

**Note:** This setting is available for CRM.Client version 1.3.1 and higher in iOS6.

---

3. For a thin connection (for example, GSM) the timeout hardly ever occurs. A timeout occurs only if time runs out before the first response package is transferred. Still, a full synchronization on a thin connection takes a long time.

---

**Note:** Do not set the timeout without limits (although possible) as the server timeout restricts the server's workload and the client timeout is important for usability.

---

## Asynchronous Server Requests

The `ClientRequestTimeout` Web Configuration parameter defines the default client timeout, see [ClientRequestTimeout](#). This timeout is also applied to a full synchronization. It is often set to several minutes to ensure, that a full synchronization can be carried out.

This timeout also defines the time span until a "Server not available" error is generated if the server does not respond to a request.

If the user connects to the server using a VPN, the scenario that a network is available but a network connection is not established hardly occurs, therefore, a long timeout interval is rather annoying.

You can define that requests are handled asynchronously. Define a timeout for the server and a retry interval. If the request has not finished processing within the server timeout, the server informs the client about this pending request. The server checks after the retry interval whether the request was processed. If the request is still pending, the server re-checks regularly until the process has finished.

To enable asynchronous server requests, set the following Web Configuration parameters:

- [System.AsyncRequestWaitTime](#). This parameter enables this feature.
- [System.AsyncRequestRetryTime](#).
- [System.DisableAsyncExecution](#).

---

**Note:** If you set `System.AsyncRequestWaitTime`, you can assign the default value (60 seconds) or even a smaller value to the `ClientRequestTimeout` parameter. All synchronous server requests (including the synchronous part of asynchronous server requests) are carried out within this time span.

---

## Synchronization Problems

Learn how to troubleshoot synchronization problems.

### Synchronization Conflicts

Learn to trouble synchronization conflicts.

Users can work offline when no server connection is available. The request is then stored on the device as an uncommitted request and the changes are applied in the local database. When a server connection is re-established, CRM.Client automatically sends the uncommitted requests to the server and updates the data in the Aurea CRM database.

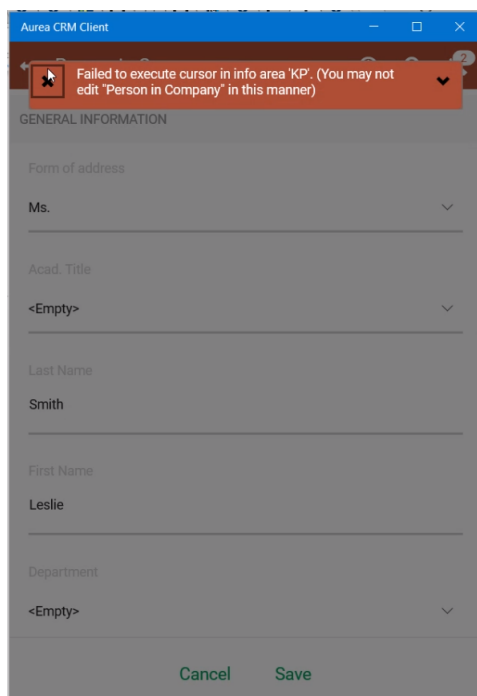
However, when a request cannot be committed the server returns an error. In this case:

1. The error is stored locally and the request is marked as blocked.
2. The requested changes are reset locally (changed data is not available anymore when searching or viewing data – both in the offline and online modes).
3. All dependent uncommitted requests (changes to the same record and new child records) are blocked and are not available for the user anymore.
4. The user is informed about the synchronization conflict and can try to fix the problem.

### Conflict Handling - Example

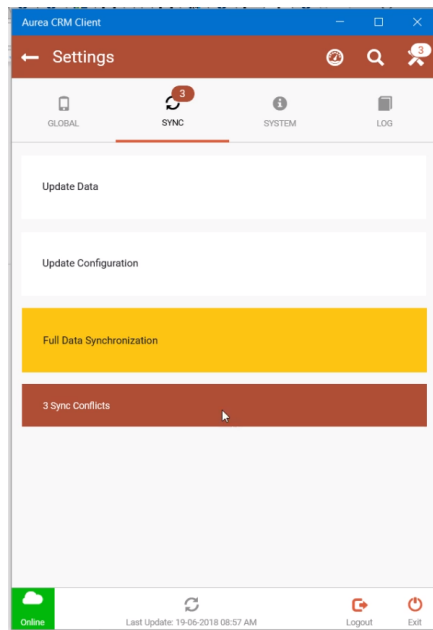
In the following example, modifying records results in synchronization conflicts and the error messages are displayed in CRM.Client:

1. The user tries to edit two person details and one person in company details for which he does not have permission.

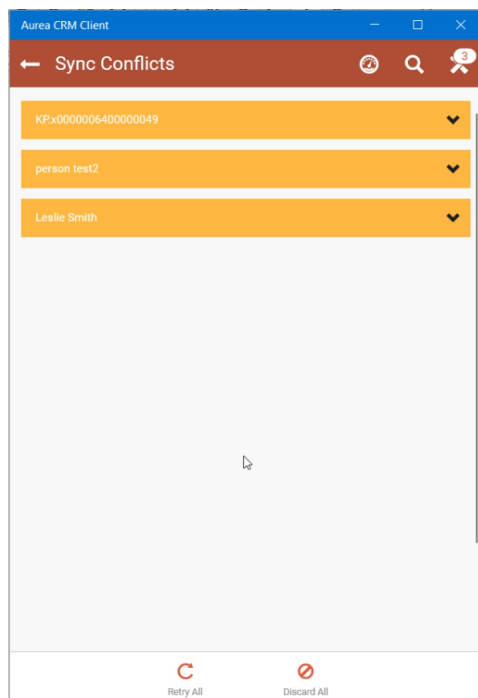


Notice that the control center icon already shows two errors.

2. If a server connection exists, an error occurs and the synchronization conflicts should be visible in the **SYNC** tab of the Control Center.

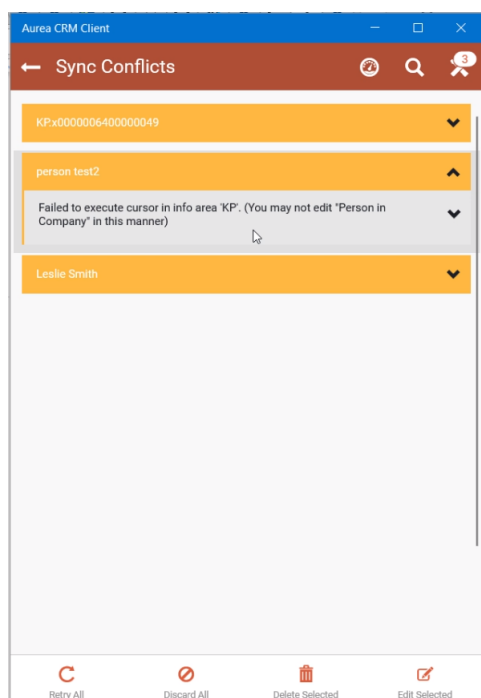


3. User can tap on **Sync Conflicts** button to see the error messages.



Notice that the **Retry All** and **Discard All** buttons are only visible, when the conflicts are not selected.

4. User can open the error message to view, edit, and discard the conflicts.



The selected conflict message is expanded and is displayed with the selection bar (yellow line on the left) and in a grey background. The buttons **Delete Selected** and **Edit Selected** are also displayed.

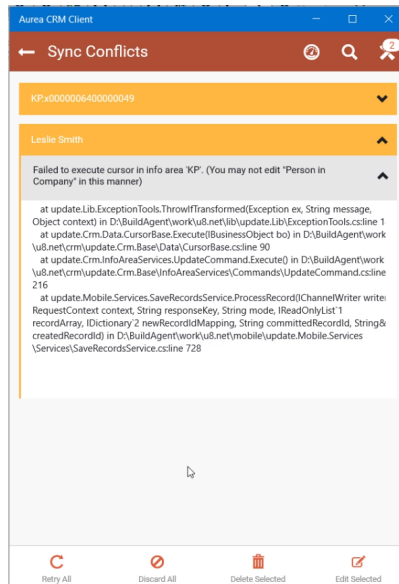
5. The user can either click on **Discard Selected** to discard the conflicting change or click on **Edit Selected** to open the record and fix the faulty data. On discarding or fixing the error the count of the synchronization conflicts would reduce.

---

**Note:** The **Edit Selected** button is available only if the conflict arises on performing a New or Edit action. For other conflict types it is greyed out.

---

6. The selected conflict message can be further expanded to see the error details, as shown in the screenshot below.



The details are displayed in a text box with a selection bar. The details are not editable but can be copied.

## Synchronization Timeout

Learn to troubleshoot synchronization timeout error.

Time-consuming single requests can occur during data synchronization.

All offline datasets in a request are queried during incremental synchronization based on a timestamp (the time of the last incremental synchronization). The server, therefore, only needs to check a small number of records and usually reacts within a short time period.

If timeout problems occur during incremental synchronizations (e.g. after many records were changed), carry out a full synchronization. If this problem occurs regularly, increase the request timeout.

Several requests are submitted one after another during a full synchronization:

- The first request transfers the complete application configuration, data model and catalog data to the client.
- The offline datasets that are synchronized are determined in CRM.designer. A request is submitted for each offline dataset, i.e. the full request timeout period is available for each offline dataset.

If problems occur during the first request, increasing the request timeout can help (although up to date no such cases are known). If the problem occurs while synchronizing one of the offline datasets, there are several options available to solve the problem. For further details, see [Full Synchronization Terminates Due to the Timeout](#) on page 105.

If you increase the timeout intervals, take into account that long synchronization requests result in a high load on the database server. Users are confronted with delays. Therefore, apply the other optimization options, in particular optimizing access rights and the database.

### Change Requests after Client Timeout

If a Save request cannot be carried out due to a client timeout, the request is stored on the device.

All change request that occur after a client timeout are not carried out until the next UpSync is carried out.

## Full Synchronization Terminates Due to the Timeout

If the full synchronization terminates, probably a timeout occurred for one of the datasets.

If this occurs, CRM.Client displays **Sync [info area]** for some time (until the request timeout occurs), then the synchronization stops. The info area displayed is usually the info area causing the problem.

To solve this problem try the following:

1. Set `LogLevel="Debug"` in the `settings.xml` file on the server and restart Aurea CRM web.
2. Synchronize again.
3. Analyze the log file (`update_net.log`) as described in the following topics.

## Analyzing the `update_net.log` Log File

Learn to analyze the `update_net.log` file.

The following topics describe how to analyze the log file.

### Finding the Cause

The `update_net.log` log file contains one line for each CRM.Client request, including the service name (in the following example "mobileService"):

```
2012.10.29 ;10:32:58 ; 0;mobileService ;Debug ;SID[41jep21jekzingzda1wtm-
lxi];Service-Name: Authenticate,user=USER, configid=1015
```

[...]

```
2012.10.29 ;10:32:59 ; 938;mobileService ;Debug ;SID[41jep21jekzingzda1wtm-
lxi];Service-Name: Synchronization,user=USER, configid=10000004
```

If multiple session IDs exist in the log file, Aurea recommends extracting the lines belonging to the same session. The timestamps on the left give information on how long each synchronization step took:

```
2012.10.29 ;10:33:39 ; 172;mobileService ;Debug ;SID[41jep21jekzingzda1wtm-
lxi];Service-Name: Synchronization,user=USER, configid=10000004
```

```
2012.10.29;10:33:48; 500;mobileService;Debug;SID[41jep21jekzingzda1wtm-  
lxi];Service-Name: Synchronization,user=USER, configid=10000004
```

```
2012.10.29;10:33:54; 156;mobileService;Debug;SID[41jep21jekzingzda1wtm-  
lxi];Service-Name: Synchronization,user=USER, configid=10000004
```

```
2012.10.29;10:34:04; 172;mobileService;Debug;SID[41jep21jekzingzda1wtm-  
lxi];Service-Name: Synchronization,user=USER, configid=10000004
```

The time between two requests is measured, this includes, the time span to execute the request on the server, to transfer the data and to process the data on the client. As client-side processing is quite quick, the difference gives a good estimation on how long processing the request takes in the network environment.

If synchronization problems occur, it is important to identify the request that causes the problem. In most cases this is the last request for the session in the log file. (Sometimes, additional requests from the same session are submitted afterwards, the runtime of the request can then be identified using the time stamps.)

You can find a semantic description of the submitted request (corresponding to the definition of the dataset and starting with `ReadByFilterMD`) under the line that corresponds to the request's start.

```
2012.10.29;10:34:05; 140;mobileService;Debug;SID[41jep21jekzingzda1wtm-  
lxi];Service-Name: Synchronization,user=USER, configid=10000004
```

```
2012.10.29;10:34:05; 63;mmObjects.Database;Info;SID[41jep21jekzingzda1wtm-  
lxi]; ReadByFilterMD (C011='', HavingOptional(C011C011)[C011='F7028 >=  
"20100101"', HavingOptional(C011C011FI)[FI='', HavingOptional(C011C011FI-  
FI)[FI='', Having:0(C011C011FIFIFI)[FI='', Having:0(C011C011FIFIFIMA)[MA='(F0  
>= "20121028" AND F0 <= "20121105")' ] ] ], HavingOptional(C011C011FIFI1)[FI='',  
Having:0(C011C011FIFI1FI)[FI='', With:0(C011C011FIFI1FIFV)[FV='F0 = "$cur-  
Rep"' ] ] ] ] ] )
```

Reformat this entry to identify what occurred:

```
ReadByFilterMD (C011='', HavingOptional(C011C011)[C011='F7028 >=  
"20100101"', HavingOptional(C011C011FI)[FI='', HavingOptional(C011C011FI-  
FI)[FI='', Having:0(C011C011FIFIFI)[FI='', Having:0(C011C011FIFIFIMA)[MA='(F0  
>= "20121028" AND F0 <= "20121105")' ] ] ], HavingOption-  
al(C011C011FIFI1)[FI='', Having:0(C011C011FIFI1FI)[FI='',  
With:0(C011C011FIFI1FIFV)[FV='F0 = "$curRep"' ] ] ] ] ] )
```

In this example, entries in the C011 info area that are linked to the FI info area are synchronized using a filter (defined in the C011 dataset). The FI records are determined according to the FI dataset definition based on the existence of MA records or Favorites.

However, the request's complexity and runtime are not immediately apparent from this information as it depends on the number of records in the corresponding tables, the defined access rights, and the size of any additional tables the access rights are applied to.

This is then followed by the SQL statement, which can be even more complicated because this statement needs to depict the access rights applied to the affected info areas (C011, FI, MA, FV) which are defined in the Rights module (`mmri.exe`).

```
2012.10.29;10:34:05; 0;update.net.core.Sql;Info;SID[41jep21jekzingzda1wtm-  
lxi];<RPACK=C011; Cur=1; SID=13A63FF0;
```

```
2012.10.29 ;10:34:05 ; 15;update.net.core.Sql ;Info ;SID[41jep21jekzingz-
dalwtm1xi]; SELECT [...]
```

## "Z Fields" in Linked Info Areas

This problem can be identified by a quite high number of follow-up requests applied to other info areas (with conditions applied to ID) following the initial SQL statement in the synchronization request:

```
2012.10.30 ;13:50:36 ; 16;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145];<RPACK=MA; Cur=1; SID=07AF61D0;

2012.10.30 ;13:50:36 ; 15;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; SELECT [...]

2012.10.30 ;13:50:37 ; 765;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; EXEC=765; Conn=2; Cur=1; Rec=1024; Ret=0; Res=1024;

2012.10.30;13:50:37; 15;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145];<READ=FI; Cur=0; SID=07AF61D0; ' ,0,0';

2012.10.30;13:50:37; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; SELECT [...] WHERE ID=?

2012.10.30 ;13:50:37 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; V1='429496729665';

2012.10.30 ;13:50:37 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; EXEC=0; Conn=2; Cur=0; Rec=1; Ret=0; Res=1;

;SID [nqgpay55rl3iil45lagq1145];<READ=KP; Cur=0; SID=07AF61D0; ' ,0,0';

2012.10.30;13:50:37; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; SELECT [...] WHERE ID=?

2012.10.30 ;13:50:37 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; V1='429496730029';

2012.10.30 ;13:50:37 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; EXEC=0; Conn=2; Cur=0; Rec=1; Ret=0; Res=1;

2012.10.30 ;13:50:37 ; 0;update.net.core.Sql ;Notice ;SID[nqg-
pay55rl3iil45lagq1145];timer=0; Ok; 0ms; SID=07AF61D0;>

2012.10.30;13:50:37; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145];<READ=FI; Cur=0; SID=07AF61D0; ' ,0,0';

2012.10.30;13:50:37; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; SELECT [...] WHERE ID=?

2012.10.30 ;13:50:37 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; V1='4294967308';

2012.10.30 ;13:50:37 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1145]; EXEC=0; Conn=2; Cur=0; Rec=1; Ret=0; Res=1;
```

You can avoid this problem by using the corresponding link fields in the configuration instead of the Z fields. These are the fields for the first name and last name from the KP info area instead of the "Person" Z field. These fields must subsequently be synchronized with the KP dataset.

This issue depends on the number of synchronized records. Reduce the offline dataset in order to execute a smaller number of additional statements.

### The Runtime of the Initial Statement is too Long

The problem can be identified by the first statement taking a long time (or because it is the last entry for this session in the log file).

Processing the request on the database server takes a long time. The runtime is determined by the dataset and access rights definition. The runtime depends on the size of the involved tables and how easy it is to depict the conditions applied to these tables. Therefore, this problem can also occur if the synchronized data volume is quite small.

### The Statement Returns too Much Data

The problem can be identified by the statement executing successfully but returning a very large amount of data - this is the case if the statement completes executing (a runtime and results set are specified) and is followed by several RPACK lines for the info area being synchronized.

```
2012.10.30 ;13:50:36 ; 16;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1l45];< RPACK =MA; Cur=1; SID=07AF61D0;

2012.10.30;13:50:36; 15;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1l45]; SELECT [...] FROM BTB70_MA [...]

2012.10.30 ;13:50:37 ; 765;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1l45]; EXEC=765; Conn=2; Cur=1; Rec=1024; Ret=0; Res=1024;

2012.10.30 ;13:50:40 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1l45];< RPACK =MA; Cur=1; SID=07AF61D0;

2012.10.30 ;13:50:40 ; 16;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1l45]; EXEC=16; Conn=2; Cmd=FNEXT; Cur=1; Rec=1024; Ret=0;
Res=1024;

2012.10.30 ;13:50:40 ; 0;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1l45];< RPACK =MA; Cur=1; SID=07AF61D0;

2012.10.30 ;13:50:40 ; 16;update.net.core.Sql ;Info ;SID[nqg-
pay55rl3iil45lagq1l45]; EXEC=16; Conn=2; Cmd=FNEXT; Cur=1; Rec=1024; Ret=0;
Res=1024;

[...]
```

This problem can also cause CRM.Client to "crash" (it is terminated by the iOS operating system because the response cannot be processed due to a lack of memory resources). This occurrence depends on a record's width (the number of fields and their length) and on the number of records (usually more than 50,000 records) in one info area synchronized on the device. Try to reduce the data volume that is synchronized to improve the performance of the client.

## Optimizing the Runtime of an Offline Dataset

Learn how to optimize the Runtime of an offline Dataset.

The following topics describe how to optimize the runtime of an offline dataset:

## Simplifying the Dataset Definition

A dataset determines all data in an info area that either is included in one of the filters specified in the dataset or is linked to at least one of the specified child records.

These conditions are included in a single SQL statement. In addition, the conditions and dependencies defined in child datasets as well as the access rights defined for all involved info areas are also included in this statement.

Combining child datasets and/or access rights definitions often results in complex database requests (which take a long time to execute).

Using filters instead of dependencies result in simpler SQL statements. Depending on the data volume you can also export all records the user is only allowed to view.

## Splitting the Dataset Definition

More than one dataset can be defined for an info area and each definition is treated as an independent request. If you define complex filters (linked using "OR") or multiple child info areas (that are also linked using "OR"), each setting can be moved to a separate dataset.

The same record can be returned by multiple datasets, but avoid large overlaps as this data needs to be transferred over the network multiple times to the iOS.

If a dataset is split into multiple datasets, child datasets are only synchronized with the original dataset. For example, if KP is synchronized for all FI, and the FI dataset is split into FI1, FI2 and FI3, both FI2 and FI3 need to be referenced by the KP dataset, or separate KP2 and KP3 datasets need to be defined for FI2 and FI3 to ensure that all KP records are transferred for all FI records.

## Simplifying Access Rights

Only a user's view access rights are relevant for the synchronization. Bottom-up access rights (where the access rights depend on sub-tables) are performance-intensive, e.g. "companies with a linked SB record where the user is entered as a rep" or "companies that had contact with the user in the last 2 years". Multiple combinations of such bottom-up access rights result need more resources to process.

Try the following approaches (which can be combined):

- **Simplification:** Edit the access rights so that they are easier to evaluate, the trade-off being a semantic change to the access rights definition. Usually, data exists that the user may not view (set A), data that the user may view but is not interested in (set B) and data that the user may view and is interested in (set C). In this case, you can try to simplify the access rights for set B. The user still cannot view set A while having a complete full view of set C.
- **Pre-triggering:** Access rights can also be simplified by using triggers that create the necessary access information upon saving the record. In an extreme case, a record could exist in a child info area for each FI record and the user is allowed to view the record. In this case, reduce the access rights to a single bottom-up condition. However, triggers need to be used to ensure that these child info areas are kept up to date.

Measures that simplify access rights not only improve the performance of CRM.Client's synchronization but also the performance of queries in all applications that access the Aurea CRM database.

## Optimizing the Database

Apply the following:

- Optimizing the database indexes (either manually or with a database tool)
- Maintenance (keeping database statistics up-to-date, avoiding the fragmentation of data files)

## Server Problems

Learn how to troubleshoot server issues.

The following error message indicates that the problem is a server issue (and not a <CRM.Client> issue): "The action could not be completed because it was rejected by the server".

1. Check the server log file (in Debug mode with WebTrace) to identify the problem (especially when the user states remarks like "but in web it is working").  
CRM.Client is not a web application, therefore functions may work in Aurea CRM web but not in CRM.Client.
2. Compare Aurea CRM web statements with those from the CRM.Client for differences. For example, some fields that are filled in web but not in CRM.Client may cause the problems.
3. Enable the request and result log for information on the complete response from the server.

## Lost Server Connection

Learn how to troubleshoot Lost Server Connection issue.

When a server connection is not available then the CRM.Client application responds as described below:

- If a server connection is lost after you login, then the CRM.Client application works in the offline mode. To get back to the online mode you have to log off and login again in the online mode.
- If a server connection is not available during login then a synchronized CRM.Client switches to login in the offline mode.