

---

# ***Aware IM***

***Version 8.5***

---

## **How To**



Copyright © 2002-2020 Awaresoft Pty Ltd

# CONTENTS

<b>INTRODUCTION .....</b>	<b>3</b>
<b>AWARE IM BASICS.....</b>	<b>3</b>
<b>FORMS AND USER INTERFACE .....</b>	<b>3</b>
<b>CHARTS.....</b>	<b>4</b>
<b>BUSINESS RULES .....</b>	<b>5</b>
<b>REPORTS AND DOCUMENTS .....</b>	<b>5</b>
<b>USERS AND LOGIN .....</b>	<b>5</b>
<b>E-MAILS .....</b>	<b>6</b>
<b>SECURITY AND PROTECTION.....</b>	<b>6</b>
<b>INTEGRATION WITH OTHER SYSTEMS.....</b>	<b>6</b>
<b>MOBILE APPLICATIONS .....</b>	<b>7</b>
<b>MISCELLANEOUS.....</b>	<b>7</b>
<b>TIPS AND TRICKS.....</b>	<b>7</b>

## Introduction

This document contains answers to some frequently asked questions about **Aware IM**. It also contains hints on how to do certain common tasks in **Aware IM**. The document has to be read in conjunction with the Aware IM User Guide.

See:

[Aware IM Basics](#)

[Forms and User Interface](#)

[Business Rules](#)

[Reports and Documents](#)

[Users and Login](#)

[E-mails](#)

[Security and Protection](#)

[Integration with Other Systems](#)

[Mobile Applications](#)

[Miscellaneous](#)

[Tips and Tricks](#)

## Aware IM Basics

[How to go about system configuration](#)

[How to start configuring the system](#)

[How to define database tables and database columns](#)

[How to define links between database tables](#)

[How to set up calculated fields](#)

[How to set up business rules](#)

[How to test the configured system](#)

[How to deploy the configured system](#)

[How to create records](#)

[How to edit records](#)

[How to look for records](#)

[How to delete records](#)

[How to change default database used](#)

## Forms and User Interface

[How to design a form](#)

[How to attach actions to buttons](#)

[How to control the order of attributes on automatically generated forms](#)

[How to initialize form controls](#)

[How to validate user input](#)  
[How to set up custom form layout](#)  
[How to set up forms with tabs](#)  
[How to use different forms for different users](#)  
[How to show related records](#)  
[How to choose values from a list](#)  
[How to set up system menu](#)  
[How to set up custom menu](#)  
[How to customize colors and fonts \(custom themes\)](#)  
[How to display a hierarchy of related records](#)  
[How to display Microsoft Outlook-like calendar](#)  
[How to synchronize with Google, Outlook, Exchange and other third party calendars](#)  
[How to show related records in different colors](#)  
[How to display different icons for different types of records](#)  
[How to display a form of a child object inside the form of the parent object](#)  
[How to display attributes in different colors](#)  
[How to display attributes as read-only](#)  
[How to make an attribute visible, read-only or mandatory on a particular form only](#)  
[How to make your application “responsive” to different screen sizes](#)  
[How to recalculate values in controls dynamically after a user changes some values](#)  
[How to show/hide/disable/enable controls dynamically after a user changes some values](#)  
[How to start processes from forms](#)  
[How to show Google Maps](#)  
[How to show thumbnails](#)  
[How to create dashboards and allow end users to customize them](#)  
[How to implement item reordering within a grid or custom query](#)  
[How to implement drag-and-drop between grids](#)  
[How to use the “Screen Tour” feature](#)  
[How to add support for electronic signature](#)  
[How to add right-to-left support](#)  
[How to embed video in a form](#)  
[How to use CSS](#)  
[How to use icon fonts to display an icon for a button](#)  
[How to display colour editor](#)

## Charts

[How to show charts](#)  
[How to add Gantt chart to your application](#)  
[How to use charts in reports](#)

## Business Rules

[What is the difference between processes and business rules?](#)

[How to define a rule](#)

[How to find a rule](#)

[How to deal with rules that need to be evaluated in a strict order](#)

[How to make sure that the action is executed only once](#)

[How to perform an action on multiple business objects](#)

[How to check whether rules are executing correctly](#)

## Reports and Documents

[How to design a report](#)

[How to produce and print out a report](#)

[How to refer to the database fields in a report](#)

[How to perform calculations in a report](#)

[How to group records in a report](#)

[How to include related records in a report](#)

[How to include an element in the report conditionally](#)

[How to print an element in different colors depending on some conditions](#)

[How to enumerate elements in a report](#)

[How to re-use a report](#)

[How to store documents and files](#)

[How to store pictures](#)

[How to do "document-merge"](#)

[How to let end users define their own reports, document templates and queries](#)

[How to implement multiple file upload for documents](#)

[How to let users edit documents](#)

[How to work with embedded spreadsheets](#)

## Users and Login

[How to set up system users](#)

[How to log into the system](#)

[How to set/change user password](#)

[How to set up users of the Configuration Tool](#)

[How to set/change password for users of the Configuration Tool](#)

[How to allow non-registered users access to the system](#)

[How to allow non-registered users to self-register](#)

[How to refer to the current user in rules](#)

[How to use Active Directory/LDAP for login](#)

[How to implement "forgotten password" functionality](#)

[How to set up Login via Facebook, Twitter and Google](#)

[How to store “session” values – values that exist only during user session](#)

## **E-mails**

[How to configure the system to send e-mails](#)

[How to send e-mail with attachments](#)

[How to configure the system to receive e-mails](#)

[How to register \(save\) incoming e-mails in the system](#)

[How to process e-mail based on its contents](#)

[How to do "mail-merge"](#)

[How to send e-mails in HTML format](#)

## **Security and Protection**

[How to give different authority to different users](#)

[How to limit access to database tables](#)

[How to limit access to records](#)

[How to limit access to operations](#)

[How to define access based on some conditions](#)

[How to get end users to define their own access restrictions](#)

[How to implement multi-tenancy](#)

[How to set up SSL](#)

## **Integration with Other Systems**

[How to use existing database tables](#)

[How to use data from several existing database tables](#)

[How to integrate credit card payment system](#)

[How to communicate with another software system](#)

[How to communicate with a hardware device](#)

[How to expose a web-service](#)

[How to expose a REST-ful service](#)

[How to consume a REST-ful service](#)

[How to use a web-service](#)

[How to register Aware IM application in Facebook](#)

[How to register Aware IM application in Twitter](#)

[How to register Aware IM application in Google](#)

## Mobile Applications

- [How to create a mobile version of a regular application](#)
- [How to display thumbnails – images with descriptions that the user can tap on](#)
- [How to display grids with column names](#)
- [How to define a slide-out application menu](#)
- [How to display mobile forms](#)
- [How to display a “gallery view”](#)
- [How to upload an image from camera or photo library of the phone](#)
- [How to display a picture full screen](#)
- [How to display a PDF document](#)
- [How to display a Google Map](#)
- [How to determine and watch location of the current user](#)
- [How to build native mobile applications](#)
- [How to remember user credentials to prevent multiple logins](#)
- [How to send push notifications to a mobile phone](#)
- [How to integrate custom Cordova components in native mobile applications](#)

## Miscellaneous

- [How to re-use configurations](#)
- [How to import/export data](#)
- [How to define import/export templates to map data](#)
- [How to set up periodic processes](#)
- [How to extend \*\*Aware IM\*\*](#)
- [How to get users to define their own workflows](#)
- [How to support offline mode](#)
- [How to count web site visitors](#)
- [How to track the activity of registered users](#)
- [How to create applications in different languages](#)
- [How to optimise application for best performance](#)

## Tips and Tricks

- [How to search for multiple objects](#)
- [What to do if an object can be related to multiple objects](#)
- [How to display a currency symbol](#)
- [How to generate unique ID for the attribute](#)
- [How to compare strings ignoring case](#)
- [How to force users to enter parameters of the search](#)

## **How to go about system configuration**

We recommend that you follow these steps when configuring your application (the order of steps is essential):

### ***1. Identify the requirements of your system***

Requirements define the problem(s) that your application will solve for your organization. All other steps are part of a solution to these problems. Without the definition of a problem there can be no solution.

### ***2. Define business objects and relationships between the objects.***

This is the key step when defining the solution. Make sure you do it right – if you do, all other steps will be easier. Ignore any user interface considerations at this stage – let **Aware IM** create the default user interface and forms for you. You can customize them later if you like. See also the “Business Objects as Carriers of Data” section in the Aware IM User Guide.

### ***3. Define business rules***

Business logic of your system must be encapsulated in business rules. Do not use processes for this. Do not worry about presentation rules (rules on how the information will be presented to the user) at this stage. See also the “Business Rules as Carriers of Business Logic” section in the Aware IM User Guide.

### ***4. Test the resulting system in the browser.***

Performing steps 2 and 3 should be enough to create the skeleton of your system. **Aware IM** will create the default user interface that will make it possible for you to test this skeleton in the browser. See also the “Testing Business Space Version” section in the Aware IM User Guide.

### ***5. Refine your system by customizing the user interface, adding reports and document templates, specifying access restrictions etc***

### ***6. Test the final system***

See the “Testing Business Space Version” section in the Aware IM User Guide.

### ***7. Deploy the final system***

See the “Publishing Business Space Version” section in the Aware IM User Guide.

See also the “Configuration Guidelines” section in the Aware IM User Guide.

## **How to start configuring the system**

1. Start the Configuration Tool. If you have just installed **Aware IM** you will see your business space with two business space versions - one in the CURRENT



- state and another one in the `NEW` state – see the “Lifecycle of a Business Space Version” section in the Aware IM User Guide.
2. Expand the node of the version in the `NEW` state (its number must be 1.1) and start defining business objects, business rules etc in the newly created version – see the “Working with Configuration Elements” section in the Aware IM User Guide.
  3. To save changes update the version. The Configuration Tool displays business space versions in the `CURRENT` state in black – see the “Updating Business Space Version” section in the Aware IM User Guide.

See also the “Getting Started with Aware IM” document.

### **How to define database tables and database columns**

In **Aware IM** you are not dealing with database tables or database columns. Instead of defining tables and columns you are defining business objects and their attributes (see the “Business Objects as Carriers of Data” section in the Aware IM User Guide). **Aware IM** automatically creates the required database tables and columns when you publish the business space version (see the “Publishing Business Space Version” section in the Aware IM User Guide) or make it available for testing (see the “Testing Business Space Version” section in the Aware IM User Guide).

### **How to define links between database tables**

In **Aware IM** you do not define links between database tables. **Aware IM** encourages you to think in terms of business objects and relationships between them rather than in terms of linking database tables through a particular column or creating special relationships tables.

To establish a relationship between two business objects, all you have to do is create an attribute in one business object with the type that refers to another business object. **Aware IM** will automatically link the required tables and/or create relationship tables if necessary.

For example, if we want to establish a relationship between the PurchaseOrder and OrderLineItem business objects, such that the PurchaseOrder may refer to multiple OrderLineItem’s, we can create an attribute in the PurchaseOrder object called “MyLineItems” of the type “OrderLineItem” and check “Multiple allowed” property of this attribute (see the “Setting Properties of Reference Attributes” section in the Aware IM User Guide).

In addition if we want the OrderLineItem object to refer to its PurchaseOrder we can create a matching attribute called “MyOrder” in the OrderLineItem object of the type “PurchaseOrder”.

See also the “Reference Attributes” section in the Aware IM User Guide.

### **How to set up calculated fields**

To get **Aware IM** to calculate the value of an attribute of a business object you have to define a business rule that will perform the required calculation. The rule should be attached to the business object that owns the calculated attribute. See the “Adding/Editing Business Rules” section in the Aware IM User Guide.

In addition you can check the “Calculated” property of the attribute. This will make sure that the attribute is read-only on the forms of the business object. See the “Adding/Editing Attributes” section in the Aware IM User Guide.

For example, if you want to define an attribute called NumberOfLineItems in the PurchaseOrder object, you can define the business rule as follows:

```
If PurchaseOrder.MyLineItems WAS CHANGED Then  
PurchaseOrder.NumberOfLineItems = COUNT OrderLineItem WHERE  
OrderLineItem IN PurchaseOrder.MyLineItems
```

### **How to set up business rules**

Most business rules should be attached to business objects. The rules will be automatically evaluated when this object is created or modified. See the “Business Rules as Carriers of Business Logic” and “Adding/Editing Business Rules” sections in the Aware IM User Guide.

### **How to test the configured system**

When you finish configuring your application (or part of it) you can test how the system will work before you deploy it for production use. To test the system you must make the business space version, containing the configuration you want to test, available for testing. To do this:

1. Make the business version available for testing – see the “Testing Business Space Version” section in the Aware IM User Guide
2. Log into the testing mode – see the “Full User Login, Interactive” section in the Aware IM User Guide.
3. Start creating business objects, reports etc – see the “Testing Mode” section in the Aware IM User Guide.
4. Make sure that business rules operate as expected – inspect the execution log for details. See the “Execution Log” section in the Aware IM User Guide.

### **How to deploy the configured system**

Any application configured with **Aware IM** is operated over the web using any web browser such as Internet Explorer or Firefox. Before the final application is

deployed the business space version containing the application configuration needs to be published – see the “Publishing Business Space Version” section in the Aware IM User Guide.

There are several deployment options available:

1. The application can be deployed on a single computer.

This option may be useful if the application will never be used in the networking environment or over the Internet. In this case the operation users may log into the application on the local machine (see the “Login” section in the Aware IM User Guide) and start using the application immediately after it has been published.

2. The application can be deployed on a local area network (LAN) within the organization.

With this deployment option the **Aware IM** server has to run on a “server” machine and the users should point their browsers to the URL of the machine where the server is running. The Configuration Tool can be started either from a “server” machine or from a remote machine across the LAN. Note that this option may also be combined with option 3. See the “Aware IM Deployment Options” section in the “Aware IM Installation Guide” for details.

3. The application can be deployed for use across the Internet.

Either your organization should have direct access to the Internet or you have to get a third party organization (“web host”) to provide this access. For details on working with **Aware IM** in the hosting environment see the “Aware IM Installation Guide”.

### **How to create records**

In **Aware IM** records are created in the operation mode. In **Aware IM** terminology creating a record is creating an instance of a business object. For users to be able to create instances of business objects the application must be configured to have the menu item of the “Create Object” type (see the “Setting Menu Item Properties” section in the Aware IM User Guide). When a user selects the menu item of this type in the Operation Mode the default form of the business object is displayed. When the user fills out the form and submits it, the instance of the business object is created in the system.

By default **Aware IM** includes the menu item of the “Create Object” type in the “Standard” menu folder. If a user selects this menu item in the Operation Mode, **Aware IM** prompts the user to select the business object to create and then displays a form of the selected object. The default menu item makes it possible to create an instance of any business object without performing any special configuration work. The disadvantage of using the default menu item is that the user has to select the business object to create every time. To avoid this you can configure a menu item of the “Create Object” type that creates a particular

business object. If a user selects such a menu item **Aware IM** will no longer ask to select the business object to create but display the form straight away.

It is also possible to configure a process that will create an instance of a business object (see the “Adding/Editing Processes” section in the Aware IM User Guide). The process can execute `ENTER NEW` action of the Rule Language, which will bring up a form of a business object for the user to fill out. The process approach should be used when the creation of an instance should be immediately followed by some other action. For the user to start the process a menu item of the “Start Process” type has to be configured.

### **How to edit records**

In **Aware IM** records are modified in the operation mode. In **Aware IM** terminology editing a record is editing an instance of a business object. In order to edit an instance of a business object the user has to find the instance first. This can be achieved by running a query – see [How to look for records](#). If a query displays its results in a standard manner the instance of a business object can be edited by pressing the Edit button (see the “Standard Form of Query Results” section in the Aware IM User Guide). This will bring up a form of the business object so that the user can change the attribute values on this form and submit it. If a query displays its result as a presentation the presentation layout may have a hyperlink defined that invokes an editing operation (see the “Setting Hyperlink Properties” section in the Aware IM User Guide). In this case clicking on the hyperlink will bring up the business object form.

It is also possible to configure a process that will find a particular instance of a business object and invoke the editing operation using the `EDIT` action of the Rule Language (see the “Adding/Editing Processes” section in the Aware IM User Guide). This process might look something like this:

```
FIND Account WHERE Account.Name='John Smith'  
EDIT Account
```

The `EDIT` action will display a form of the business object and wait for the user to change the values and submit the form. For the user to start the process a menu item of the “Start Process” type has to be configured.

### **How to look for records**

In **Aware IM** terminology looking for records is looking for instances of a business object. A user can find instances of a business object in the operation mode by running a query. There are several ways of running a query in the operation mode. See the “Running Queries in the Operation Mode” section in the Aware IM User Guide for details.

## **How to delete records**

In **Aware IM** terminology deleting records is deleting instances of a business object. To allow the user to delete instances of a business object in the operation mode either a process that finds an instance of a business object and uses the `DELETE` action to delete it, has to be configured (see the “Adding/Editing Processes” section in the Aware IM User Guide), or an operation of the Delete Object type has to be defined for the object (see the “Adding/Editing Form Menu” section in the Aware IM User Guide).

## **How to change default database used**

Select “Database” from the Settings menu in the **Aware IM** Control Panel. In the Database Settings dialog choose the database engine you want. Set the properties of the database engine. Click OK and restart **Aware IM**.

## **How to design a form**

There are three types of forms used in **Aware IM**:

### ***Object Form***

This form is used to enter or change values of attributes of a business object. **Aware IM** generates the default layout of a form automatically for every business object. You can control how this is done - see the “Business Object Forms” section in the Aware IM User Guide for details.

### ***Presentation***

Unlike the object form, which is used to enter or change values of attributes of a business object, presentation is used to *display* these values (usually after a user performs a search by running a query). **Aware IM** supplies the standard form that displays the results of any query but you can also design your own presentation form using the Report/Presentation Designer. See the “Data Presentation” section in the Aware IM User Guide for details.

### ***Visual Perspective***

This form represents a top-level screen that users see when they log into the application. The screen usually contains the system menu, status information, company logo etc. **Aware IM** generates the default visual perspective, but you can customize the default perspective and/or supply your own. It is also possible to define several visual perspectives and switch between them. See the “Adding/Editing Visual Perspectives” section in the Aware IM User Guide for details.

## **How to attach actions to buttons**

There are three types of forms used in **Aware IM** – object forms, presentations and visual perspectives (see [How to design a form](#)).

### **Object Form**

Most of the time you do not need to define buttons for the object forms. **Aware IM** automatically generates the Save and Cancel buttons, that allow submitting or cancelling the changes to the business object, for these forms. **Aware IM** also automatically generates the button in the form caption and/or hyperlinks that allow invoking operations with a business object being edited – see the “Panel Operations” section in the Aware IM User Guide for details. You can, however, add your own custom buttons that perform other operations (such as start a process) from a form. To do this you need to define an HTML cell in the form and add an HTML button element that starts a process – see Adding/Editing Business Object Forms section in the User Guide (editing of HTML cells).

### **Presentation**

You can define hyperlinks in your presentation that will invoke certain operations, such as start a process, run a query etc. See the “Hyperlinks” section in the Aware IM User Guide for details.

### **Visual Perspective**

Usually a visual perspective contains the system menu, which is automatically generated by **Aware IM**. You can configure the names of the menu items and the types of operations they invoke as part of the visual perspective configuration. See the “Setting Menu Item Properties” section in the Aware IM User Guide for details.

You can also supply your custom menu instead of getting **Aware IM** to generate its own. See [How to setup custom menu](#).

### **How to control the order of attributes on automatically generated forms**

When **Aware IM** generates an automatic form for a business object by default it generates controls for all attributes in the order that the attributes have been added to the business object. You can get **Aware IM** to generate the controls only for some attributes and specify the desired order of the controls by specifying the attributes and their order and layout in the Form Section Dialog. See the “Adding/Editing Form Sections” section in the Aware IM User Guide for details.

### **How to initialize form controls**

When the user is entering attribute values of a new business object using an object form you can get **Aware IM** to initialize controls in the form with some values. In order to do this you need to configure the initial value for the attribute represented by the control you want to initialize. See the “Common Properties” section in the Aware IM User Guide for details on how to do this.

If initialization of an attribute depends on some complex conditions you can define a business rule that sets the value of the attribute based on these conditions and declare that this rule should be used during form initialization. See the “Initialization Rules” section in the Aware IM User Guide.

### **How to validate user input**

Certain validation rules can be specified when you define attributes of an object. For example, you can specify that the attribute may take only certain values (Choice list) or that the attribute value may only be within the given range (Value Range). See the “Common Properties” section in the Aware IM User Guide for details on how to do this.

If you want more complex validation of the attribute values then you can attach a rule to the object that checks the value of the attribute. Use `REPORT ERROR` action to indicate the error action. When this action is executed **Aware IM** will abort any changes done to the object and display the specified error message to the user suggesting to re-enter the invalid value. For example:

```
IF Client.Spouse IS DEFINED AND
Client.MaritalStatus='Single' Then REPORT ERROR 'Marital
status cannot be single when spouse is defined"
```

### **How to set up custom form layout**

If you want references of the object to be displayed not on tabs inside the form of the object but as separate panels altogether you can use the `DISPLAY LAYOUT` action.

To use the `DISPLAY LAYOUT` action to provide custom layout of attributes and references do the following:

1. Define a process that would show the form of the object (the object should be passed as process input)

```
VIEW MyObject
```
2. Define queries that would show references of the object (the queries should assume that the object is in the context, for example:

```
FIND MyReference WHERE MyReference IN
MyObject.References
```
3. Define a visual perspective with a single tab inside the main frame
4. Define content panels of this tab to show a form of the object in one content panel and references of the object in the other content panels. The panel showing the form should use the process defined in 1.; and panels for references should use queries defined in 2.
5. Define a process that would display the layout:

```
FIND MyObject WHERE ...
DISPLAY LAYOUT MyTab FROM_VP MyPerspective
```

## **How to set up forms with tabs**

If you want to design an object form that has tabs you need to define two or more form sections for the form – every form section will be represented by a tab. The name of the form section will be the name of the tab. You should also set the navigation style of the form to be “Random”. See the “Form Sections”, “Adding/Editing Form Sections”, “Choosing Form Section Navigation Style” sections in the Aware IM User Guide.

## **How to use different forms for different users**

You may want to display the information differently for different users – for example, some information should be visible to some users, but not others. You may also want to display the information using different colors and fonts. **Aware IM** allows displaying different forms for different categories of users based on their access level. There are three types of forms used in **Aware IM** – object forms, presentations and visual perspectives (see [How to design a form](#)).

### ***Object Form***

When configuring a business object form you can specify the access level(s) that the form will be used for. Therefore if you want **Aware IM** to display different forms for users of different access levels you can configure several forms for the business object and specify the access level(s) that each form will be used for. See the “Adding/Editing Forms” section in the Aware IM User Guide for details.

### ***Presentation***

Presentation is usually used to display the results of a query in a non-standard fashion. Therefore you can configure different presentations and different queries (each displaying its own presentation) and configure the application to run the query appropriate for the access level (for example, a menu item in a visual perspective for one access level could run one query, whereas the menu item in a visual perspective for a different access level could run another query) - see the “Configuring Queries” section in the Aware IM User Guide. Alternatively you can design conditional elements in your presentation, the conditions of which will check the access level of the current user (this is recommended only if the differences in presentations for different users are small). The example of such a condition:

```
If LoggedInRegularUser.AccessLevel = 'Operator' Then ...
```

See also the “Editing Conditional Element” section in the Aware IM User Guide.

### ***Visual Perspective***

You can specify the access level that the visual perspective will be used for as part of the configuration of the visual perspective. Therefore you can configure



different visual perspectives for different access levels. See the “Adding/Editing Visual Perspectives” section in the Aware IM User Guide.

### **How to show related records**

You can show the values of attributes of the business objects related to a particular business object on the forms or presentations of this object. For example, if a PurchaseOrder object is related to an OrderLineItem object via the multiple relationship you can show which order line items are related to a particular purchase order on the form or presentation of this order.

#### ***Object Form***

**Aware IM** will automatically show the related objects on the object form if you include the reference attribute that represents the relationship in the list of attributes to be displayed on the form. You can control how the related objects will be shown – which attributes of the related objects will be displayed, whether they will be sorted etc. See the “Presentation Options For References” section in the Aware IM User Guide for details.

Also if an object is related to another object via a single relationship you can display the value of just one attribute of the related object on a form. In order to do this you have to define an attribute of the Shortcut type that will point to a particular attribute of the related object. See the “Setting Properties of Shortcut Attributes” section in the Aware IM User Guide.

#### ***Query Results***

To display attributes of related objects on the query results form you have to define an attribute of the Shortcut type that will point to a particular attribute of the related object (see the “Setting Properties of Shortcut Attributes” section in the Aware IM User Guide) and include this attribute in the list of attributes to be displayed by a query (see the “Adding/Editing Queries” section in the Aware IM User Guide).

#### ***Presentation***

To display the values of attributes of the related object on a presentation you have to define a sub-presentation element in your presentation. The sub-presentation element must refer to the presentation of the related object and the query of the sub-presentation element must find the related objects, for example:

```
FIND OrderLineItem WHERE OrderLineItem IN  
PurchaseOrder.MyLineItems
```

See also the “Editing Sub-report/Sub-presentation Element” section in the Aware IM User Guide.

## **How to choose values from a list**

If you want a user to pick attribute values from a list instead of (or in addition to) entering them manually while entering or editing a form of a business object you have to do the following:

- If the contents of the list are static, i.e. the list always contains the same values, then you have to configure the attribute to have “static” choices and define the values of these choices. While defining choices you can also specify whether other values are allowed or not. See the “Common Properties” section in the Aware IM User Guide for details.
- If the contents of the list are dynamic, i.e. you cannot define the values of the list up-front at the configuration stage, then select the “dynamic choices” radio button and define a query to run to determine choices. Alternatively, the list can represent a related business object and the attribute can be a reference to this business object. For example, if you want a user to choose the type of the car from the list on a form of the Car object (and you do not know all possible car types at the configuration stage), then you need to configure the CarType object and get the Car object to refer to the CarType object through the reference attribute called “MyCarType”. Also when defining presentation properties of the MyCarType attribute you have to specify that the attribute should be represented by the drop down list rather than by the relationship table (see the “Presentation Options For References” section in the Aware IM User Guide).

## **How to set up system menu**

**Aware IM** automatically generates the menu for the configured application. The default menu includes very generic operations that allow you to create any business object, run any query or start any process. In most cases you would probably like to customize the generic menu and include the menu items that create specific objects, run specific queries or start specific processes. For details on how to do this see the “Setting Menu Item Properties” section in the Aware IM User Guide.

You can also provide your custom menu in addition to or instead of the menu generated by **Aware IM**. See [How to set up custom menu](#).

## **How to set up custom menu**

If you do not want **Aware IM** to generate the menu for the configured system you can provide your own custom menu (or you can combine both menus). To do this:

1. Prepare your own HTML page with your own menu. The menu should include links invoking **Aware IM** operations. The full list of the URL's corresponding to the **Aware IM** operations is provided in the Appendix D of the Aware IM User Guide.
2. Import this page as the Main Page of the visual perspective – see the “Setting Main Page Properties” section in the Aware IM User Guide.
3. If you do not want **Aware IM** to generate its own menu, specify that the menu frame should not be present – see the “Working With Visual Perspective Editor” section in the Aware IM User Guide.

### **How to customize colors and fonts (custom themes)**

Colors and fonts of your application can be specified when defining visual perspectives. A particular set of colors and fonts is called a “theme” or “skin”. You can choose different pre-defined themes for your visual perspective and you can also define your own themes.

**Aware IM** uses themes of the Kendo UI library (<http://www.telerik.com/kendo-ui-html5-framework>). Kendo UI requires that you provide 3 files for your theme – the theme file itself, the “common” file and the “data visualisation” file, for example:

- kendo.common-bootstrap.min.css
- kendo.bootstrap.min.css
- kendo.dataviz.bootstrap.min.css

In addition **Aware IM** provides its own CSS file -

AwareIMRoot/Tomcat/webapps/AwareIM/aware\_kendo/resources/css/aware.css (where AwareIMRoot is the root directory of your **Aware IM** installation). The file contains **Aware IM**-specific settings common to all themes. If you want to override any of these settings in your custom theme, you need to provide the CSS with the overrides, for example aware\_nova.css (for the Nova theme)

So you need to provide all of these files and then to integrate them into **Aware IM** you need to modify the file theme.props located in the AwareIMRoot/bin directory to add the new theme-specific entry starting with the prefix “Theme”, for example

```
ThemeMy=Bootstrap;KendoUI/styles/kendo.common-  
my.min.css;KendoUI/styles/kendo.my.min.css;KendoUI/styles  
/kendo.dataviz.my.min.css;aware_kendo/resources/css/aware  
_my.css
```

The name of the entry should start with the prefix “Theme” followed by any unique identifier. The value of the entry should start with the name of the theme as it will appear in the Configuration Tool (My Theme), followed by the semi-colon and then followed by the relative path to theme-specific CSS file.

After you re-start Aware IM the name of your theme will become available in the Themes dropdown in the Configuration Tool.

### **How to display a hierarchy of related records**

You can display related records in a tree-like hierarchical fashion. For example, a design component (DesignComponent object) may consist of sub-components; each sub-component may consist of other sub-components etc. To display a hierarchy of such components in a tree you need to define a reference attribute (called for example, SubComponents) in the DesignComponent object that refers to multiple other components (also DesignComponent object) – see [How to define links between database tables](#).

Then open the SubComponents attribute for editing, go to the Presentation dialog and choose “Tree” from the “Display as” drop down. For more details see also the “Presentation Options for References” section in the Aware IM User Guide.

Another way of displaying a hierarchy is to display a standard grid of parent records first (Query using the Standard form) and then define a query that would show children records for a selected parent. Then set up the parent query to display the children query when a row is expanded (see Adding/Editing Queries section in the User Guide). For each children query you can also display an operation to show its children when a row is expanded.

### **How to display Microsoft Outlook-like calendar**

To display your appointments, activities, meeting request in a calendar fashion similar to Microsoft Outlook do the following:

1. Use the “Edit/Add Appointment Object” command in the application menu of the Configuration Tool to add the skeleton of the Appointment object to your configuration. Add your own attributes to the Appointment object if necessary.
2. Define a query that will query on the Appointment object. In the “Display Results” section of the query choose “Calendar” option. The calendar will be shown when the query is run.
3. You can also define a reference attribute that will point to the Appointment object. For example, you can define the Appointments attribute in the Doctor object. Go to the Presentation dialog of this attribute and choose “Calendar” from the “Display As” drop down. The calendar will be shown on the form of the Doctor object.

For more details see the “Calendar Presentation of Query Results” and “Presentation Options for References” section of the Aware IM User Manual.

### **How to synchronize with Google, Outlook, Exchange and other third party calendars**

You can synchronize the appointment records stored in your Aware IM system with third party calendars, for example Google Calendar, Outlook, iCloud and so on. Synchronization with third party calendars in Aware IM is performed using Cronofy API, so before you can use synchronization you need to create an account with Cronofy (<https://www.cronofy.com/>) and create an application for your Cronofy account. Once you create an application with Cronofy you will be given “Client ID” and “Client Secret”. Write them down as you will need them in Aware IM. You can also see a full list of calendars that are supported at the Cronofy web site.

To support calendar synchronization in Aware IM you will need to do the following:

1. Create a query that will display “Appointment” objects as a calendar (see the previous section)
2. Define a Panel Operation for your query. Select the type of the operation to be “Synchronize with Calendar”.
3. Click on the Settings button to define some properties of this operation. You will then need to enter the Client ID and Client Secret that you have obtained from Cronofy.
4. If your “Appointment” object supports “appointment owners” (for example, a list of attendees of an event or a list of people responsible for the task) you can get Aware IM to read the information about event attendees and automatically associate them with objects in your application. To do this:
  - a. Define an object representing an “event attendee”.
  - b. Provide a relationship between an “Appointment” object and the “Attendee” object – it has to be a multiple reference inside the Appointment object
  - c. When specifying settings of the synchronization operation tick the checkbox that your appointments (events) support attendees.
  - d. Select the reference attribute defined at b)
  - e. Define how Aware IM should map attendees read from a third party calendar to the “attendee” object defined in your application – it can map them by their email address or display name or both.

At runtime the operation will be represented by a button displayed above or below the calendar. When the users press this button Aware IM will ask the user to select the calendar to synchronize with and whether perform a one-way or two-way synchronization. You can perform synchronization multiple times – in this case Aware IM will only deal with changes that happened to occur in both calendars since the time of last synchronization.

Note that there is also a type of Panel Operation that synchronizes with a Google Calendar only. This operation type has now been superseded with a more generic operation and will be eventually phased out. However it can still be used now to synchronize with a Google Calendar.

### **How to show records in different colors**

You can display records in different colors depending on some condition. For example, you might want to show overdue loans of a library member in red while showing non-overdue loans in black. To do this:

1. Open the required reference attribute for editing (for example, “Loans” attribute of the “Member” object)
2. Go to the Presentation dialog and click on the “Item Rules” button
3. Specify presentation rules using rule table, for example:  
If Loan.DueDate < CURRENT\_DATE Then  
Display in red

For more details see the “Presentation Options for References” section of the Aware IM User Manual.

### **How to display different icons for different types of records**

You may want to show an icon next to a particular record when a record is displayed in different lists such as results of a query or a list of children on a form of the parent object. You may want to show different icons depending on the value of some attribute in the record – for example, show different icons depending on whether an e-mail is read or un-read. To do this you need to specify “presentation rules” in the presentation of the attribute. The rules specify which icons will be displayed under which conditions – for example if the value of the “State” attribute is “READ” display “read” icon, if the value is “UNREAD” display “unread” icon. Read more about presentation rules in the “Common Properties” section of the Aware IM User Guide.

### **How to display a form of a child object inside the form of the parent object**

Define a reference attribute to the child in the parent object. Go to the Presentation dialog of this attribute, tick “Allow Inline Editing” checkbox, select the “Show Form for Each Row” radio button and then click on the “Details” button to select the form of the child object to be shown.

### **How to display attributes in different colors**

If you want to display certain attributes of an object on a form in different colors you can use HTML cells in a form layout and reference an attribute inside the cell. For example, if you want to display attribute “MyAttribute” of the object “MyObject” in red define the following HTML cell in your form layout:

```
<font color='#ff0000'><<MyObject.MyAttribute>></font>
```

Note that with the above method you can only show attributes as read-only.

You can also set attribute styles when defining presentation properties of an attribute – see the “Common Properties” section under the “Add/Edit Attributes”

section in the User Guide. With this method the attribute does not need to be read-only. You can set styles for input controls or labels or both. The advantage of this method is that you can define conditions for displaying values and labels in different colors.

If you want to show specific attribute values in different colors in queries you can define styles for the attribute as described above and then indicate that you want to use these styles in a particular query. To do this click on the “Details” button in the Display Results section and then click on the “More” button located next to the attribute entry and tick the “Use Attribute Styles” checkbox.

### **How to display attributes as read-only**

*Aware IM* will automatically display “calculated” attributes and attributes protected by the PROTECT rules as read-only, but it will display them in a read-only text box. If you want to get rid of the text box you can use HTML cells in a form layout and reference an attribute inside the cell. For example, if you want to display the attribute “MyAttribute” of the object “MyObject” define the following HTML cell in your form layout:

```
<<MyObject.MyAttribute>>
```

Note that you can also use expressions inside tags and use the HTML tags to display attributes as bold, colored, italicised etc.

### **How to make an attribute visible, read-only or mandatory on a particular form only**

Normally you define whether an attribute is visible or read-only using dynamic business rules and you define whether an attribute is mandatory by ticking the “Required” property of the attribute. This makes the attribute visible, read-only or mandatory on all forms. There are scenarios, though where you want to make the attribute visible, read-only or mandatory on a particular form only. To do this:

1. Open the form section that has the attribute using the Form Designer
2. Select the required attribute
3. Click on the “Presentation rules” property.
4. Define the appropriate presentation rule (conditional or unconditional) – select ACCESS as element type to make the attribute visible or read-only (specify this in the Element dialog) or select REQUIRED as element type to make the attribute mandatory

### **How to make your application “responsive” to different screen sizes**

This is explained in detail in the video tutorial available here:

[https://youtu.be/6\\_yJ4Xy6bU](https://youtu.be/6_yJ4Xy6bU)

### **How to recalculate values in controls dynamically after a user changes some values**

Sometimes it may be necessary to dynamically recalculate values in controls of a form after a user has changed values in other controls. For example, let's say we have a form of the object "MyObject" showing 3 attributes – "Value1", "Value2" and "Sum". We want to dynamically show the value of the attribute Sum as Value1 + Value2. We want the system to show the sum immediately after the user changes the values of Value1 or Value2. To do this we need to define a business rule:

```
MyObject.Sum = MyObject.Value1 + MyObject.Value2
```

This is not different from any other business rule. The trick here is when defining the rule to go to the "Advanced" tab and tick the "Use rule in dynamic recalculation on forms" checkbox. If this is ticked **Aware IM** will automatically fire the rule when any field that the rule depends on loses focus, and display the result.

### **How to show/hide/disable/enable controls dynamically after a user changes some values**

This is a variation of the previous item. Sometimes it may be necessary to dynamically show/hide controls on a form after a user has changed values in other controls. For example, let's say we have a form of the object "MyObject" showing 2 attributes – "Value1" and "Value2". Value1 is of the Yes/No type and is shown as a checkbox. We want to dynamically show or hide the value of the attribute Value2 depending on whether the checkbox is ticked. To do this we need to define a business rule:

```
IF MyObject.Value1 = 'No' Then READ PROTECT MyObject.Value2  
FROM ALL
```

This is not different from any other business rule. The trick here is when defining the rule to go to the "Advanced" tab and tick the "Use rule in dynamic recalculation on forms" checkbox. If this is ticked **Aware IM** will automatically fire the rule when any field that the rule depends on loses focus, and show/hide the appropriate controls.

### **How to start processes from forms**

It is sometimes necessary to have a button or a hyperlink on a form of the object that starts a process using the current instance of the object as a parameter. To implement such functionality you need to define an HTML cell and place a button or a hyperlink inside this cell (this is explained in detail in the Editing Form Section Layout section in the User Guide).

### **How to show Google Maps**

You can display Google Maps in the following places in the system:

- In static HTML of a content panel of a visual perspective
- In object forms



- In object presentations

A Google map shown on a form can only show one address and this address must be stored in some attribute of the object. A Google map shown in a visual perspective or a presentation can show multiple. Again an object that the query queries on must have the attribute that stores the address to be shown on a Google map.

To define a Google map on a form define a form layout cell of the “Google Map” type – see the “Editing Form Sections” section of the User Guide. To define a Google Map on a presentation select the Google Map element in the Report Designer when defining a presentation for an object. Then define a query that would display this presentation. If a query finds multiple instances of an object, multiple addresses will be shown on the Google Map.

Defining a Google Map in a content panel of a visual perspective is explained in detail in the “Defining Content Panel with Static HTML content” section of the User Guide.

You can show colored areas on Google Maps using the so-called “polygon” feature. Each area must be represented as a collection of map points in the KML format. An example of the polygon specified in the KML format is provided here: [https://developers.google.com/kml/documentation/kml\\_tut#polygons](https://developers.google.com/kml/documentation/kml_tut#polygons)

To display polygon areas on a Google Map you need to do the following:

- Define a business object representing the area. This object can have any attributes you like, but it must have an attribute that will store the KML representation of the polygon as described in the example above.
- Define a query that will retrieve instances of this object. Note that:
  - You can define item rules on this query to specify the background color of the area
  - You can define “Mouse Over” options to specify what happens when the user moves the mouse over the area
  - You can define operations to specify what happens when the user clicks on the area
- When specifying Google Map options provide the query above and identify the attribute of the “Polygon” object that stores the KML data of the polygon.

Note: Some usages of Google Maps require you to create an account and specify a special key. If you need to provide a key for your Google Maps account you can do this in the predefined GoogleMapsKey attribute of the SystemSettings object.

### **How to show thumbnails**

To show pictures as thumbnails you need to have an attribute of the Picture type defined in some business object. Then you can define a query that would show this attribute in the query results. The presentation properties of the Picture attribute allow you to specify scaling options for pictures displayed by queries.

Perhaps a better way to display thumbnails is to use the custom presentation for queries and select the “Thumbnails” data template. This is described in the “Displaying Query Results” section of the User Guide. Using this option you can also display descriptions of the picture underneath or above the picture (or both).

### **How to create dashboards and allow end users to customize them**

To create a dashboard you need to define a visual perspective with at least one tab containing content panels. This is explained in the “Adding/Editing Visual Perspectives” section of the User Guide. You can define content panels containing results of queries, processes or HTML. Content panels can be laid out according to one of the available layouts. You can also allow end users to edit the initial layout – switch visibility of individual panels as well as move or resize them. This is explained in detail in this [video tutorial](#).

### **How to implement item reordering within a grid or custom query**

**Aware IM** supports reordering of items using drag and drop within grids (standard query results) and custom query views (custom query results).

If you want to support item reordering within grids or custom views you need to define a process that would handle the reordering of the items. When the user performs a “drop” she always does it on some node in the grid or custom view. This node always represents some instance of a business object. We will call it the “drop instance”. The node being dragged also represents some instance of the business object. We will call this instance the “drag instance”.

The process implementing the drop must have the object representing drag and drop instances as its input. Both drag and drop instances will be in the Context of the process. The “drop” instance will always be the first in the Context and can be referred to with the “This” prefix. The drag instance will be the second in the Context and can be referred to with the “That” prefix. The rest is up to the process – it can do whatever it wants with the drag and drop instances. For example, if we are reordering a list of tasks and the Task object has the SequenceOrder attribute defining the order of the task in the list we could write our process to do the following:

```
ThatTask.SequenceOrder = ThisTask.SequenceOrder + 1
```

Once you have defined processes to handle item reordering you can start configuring it for those components that need to support it. Go to the “Drag-and-

drop/Reordering” property of standard or custom queries, select the “Allow Reordering” radio button and specify the process that handles item reordering.

### **How to implement drag-and-drop between grids**

A record can be dragged from one grid represented by a standard query to another grid represented by some other standard query. Note that this will only work if both queries query on the same business object.

In order to support this, you need to define a process that will be run when a record is dropped. For example, if you have a query showing open issues and a query showing closed issues and you drag an issue from the “open issues: grid to the “closed issues” grid a process will need to change the status of the issue to Closed. A record being dragged is automatically given as input to the process, so the process has to declare the Issue object as input and change its status

```
Issue.State = 'Closed' .
```

You then need to open the “Drag-and-drop/Reordering property of the query, whose records are being dragged and tick the “Allow dragging records of this query” checkbox. Then you need to go to the same property of the target query and tick the “Allow dropping records of the same type” checkbox and then specify the process to be run.

### **How to use the “Screen Tour” feature**

The “Screen Tour” feature allows configurators to add explanation to the elements of a form, query or visual perspective, so that the user can understand the screen better. The explanation of the element is displayed to the user at the bottom of the screen with the marker pointing to the element of the screen being displayed. The user can jump between the elements of the screen and Aware IM automatically scrolls to the element being explained and displays the marker for the element. This section explains how a configurator can add tours to forms, queries and visual perspectives.

Adding tours to forms and queries is reasonably straightforward. Before you add a tour you need to assign unique “id’s” to the elements that you want to mark on a form or a query. You will then refer to these id’s when you define a tour. You can assign id’s to the form cells, query columns, operations on a form or query, form sections. Just make sure that these id’s are unique within a form or a query.

Once id’s are assigned you can click on the Tour button on the form or query and define “tour sections”. When you define a tour section you need to provide the id of the element that the section applies to and the explanation of the section. At runtime Aware IM will mark the element identified with the provided id and display an explanation of the element at the bottom of the screen. If an element is a form cell Aware IM will show a marker next to the cell, if it’s a form section Aware IM

will show the marker next to the tab of the section, if it's an operation it will put the marker next to the button of the operation and so on.

Note that tour sections are ordered – Aware IM will show the tour by marking the first section and then the user can jump to the next section in the order.

Adding tours to visual perspectives can be slightly more complicated. Just like with forms and queries you can assign unique id's to the elements of a visual perspectives – tabs and content panels (the latter is done in the Other dialog of the content panel). Aware IM will then mark the corresponding tabs and panels on the screen. However, visual perspectives can also include forms and queries for which you have already defined their own tours. When defining a tour for a visual perspective you have an option of including these tours in the tour of the visual perspective. In this case refer to the id of the enclosing content panel, tick the “Include tour of the element shown in the content panel”

Note also that visual perspectives, forms and even queries can include HTML sections. If you want to mark an element inside such an HTML section you can add id to the corresponding HTML element, for example:

```
<div id='some unique id' >
```

You can then refer to this id just like you refer to the id of other elements of the visual perspective. The only difference is that you need to specify this id explicitly rather than choose it in the drop down of the defined id's.

For an example of using the Tour feature see the Photo Album sample application (visual perspectives for Author and Guest access levels)

### **How to add support for electronic signature to your application**

If you want end users to sign electronically using the mouse (or fingers or stylus pens on mobile devices) you can add support for the electronic signature to your applications. To do this you need to add an attribute of the Picture type to one of your objects. An electronic signature will be stored in this attribute as a PNG file. When defining properties of this attribute, tick the “Represents signature” checkbox. When you add such attribute to your form Aware IM will automatically show a signature component that will allow users to sign electronically.

### **How to add right-to-left support**

To enable RTL-support for a user make sure that the value of the RTL attribute in the user is set to Yes. The `RTL` attribute is a predefined attribute that is added by Aware IM automatically to all members of the `SystemUser` group. You can set the value of the `RTL` attribute by rules or explicitly from the user interface when the user record is created.

### **How to embed video in a form**

You will need to define an attribute of the Document type in some object and then add an HTML cell to your form with the following contents:

```
<div style="width:400px;height:400px"><<MyObject.DocumentAttribute>></div>
```

The div element is just to give width and height to the video. The actual video is embedded by a reference to the document attribute (MyObject.DocumentAttribute)

### **How to use CSS**

If you are familiar with the CSS technology then you can customize and enhance the look and feel of your **Aware IM** application. You can also improve “responsiveness” of your application by providing media-specific definitions for your screen elements.

**Aware IM** lets you assign CSS classes and styles to most of the widgets it uses – grids, calendars/schedulers, forms, visual perspectives, tabs and so on. You can, therefore, use custom fonts, colors, visual effects for specific queries, visual perspectives or forms. In addition you can refer to your specific CSS classes when defining your custom HTML fragments – in forms, in content panels of your visual perspectives, in custom panel headers, etc. See the Sales Portal and CRM sample application for the examples of using CSS classes in an application. Many queries and forms in these applications use custom CSS classes.

You can create one or more CSS files and put your class definitions there. You can then put your file(s) into the

AwareIM/Tomcat/webapps/AwareIM/Custom/CSS folder. Or you can create a sub-folder within this folder and put your CSS file(s) there. The difference is that CSS files placed into the root CSS folder are available to ALL your applications, whereas if you put the file(s) in the sub-folder they become available to a particular application only. In this case you have to make sure that the name of the sub-folder is exactly the same as the name of the corresponding business space.

Placing the files in these folders is all you have to do – at runtime **Aware IM** will automatically upload and use the files from these folders.

### **How to use icon fonts to display an icon for a button**

When you define icons for your buttons (in operations, menu commands, forms etc) **Aware IM** offers you a choice to provide the file of the icon or refer to the CSS class definition of the icon. In this section we will explain how to do the latter.

When you refer to a CSS class for the icon definition you normally refer to the icon defined in a special “icon font” that contains a large collection of such icons. The advantage of such fonts is that you can display the icon in different colors and sizes. To refer to a particular icon you need to know the name of the CSS

class that corresponds to the icon. The makers of the icon font usually provide a sheet with all icons it has and the corresponding CSS classes.

By default **Aware IM** uses icons from the popular “Font Awesome” (the cheat sheet of this font is located here: <https://fontawesome.com/cheatsheet>)

So anywhere in Aware IM you can refer to the icons of this font. For example, to refer to the “edit” icon you can specify `fa fa-edit` as the name of the CSS class.

What if you want to use icons from some other font? In this case you need to provide the definition of this font in your own CSS file (see [How To use CSS](#)). Or you can just add the CSS file provided by the font maker to the `AwareIM/Tomcat/webapps/AwareIM/Custom/CSS` folder

Icon colors can be specified directly in the dialog. If you want to define icons in different sizes you need to provide your own CSS class definition of these sizes in your custom CSS file and mention them when you refer to the icon. For example,

```
.crm-large-font {  
    font-size: 26px;  
}
```

And then refer to these styles and the icon itself like so:

```
crm-large-font fa fa-edit
```

### **How to display colour editor**

Sometimes you need to store a colour, which you can edit using a colour editor. To do this define an attribute of the Plain Text type in an object – this attribute will store a textual value of the colour (need about 10 characters). Change the “Content” property of the attribute to specify “Colour”. When you do this the attribute will be displayed using a Colour Picker widget on forms. If this attribute is used in queries it will be displayed as a box with the background colour stored in the attribute. If you want to display textual values go to the “Display Format” property of the query column and select “Text”.

### **How to show charts**

The best way to show charts is to define a query that would retrieve data for charts and then select “Chart” as the presentation option for this query. This is explained in the “Display Query Results” section of the User Guide. The query should find instances of objects whose attributes provide data values for the X and Y-axis of the chart. Charts displayed this way will be dynamic – the user will be able to see the values of the points on the chart as he hovers the mouse over the data. He will also be able to click on the chart and start some operation (that you can define).

It is also possible to define charts in presentation and reports. Such charts will be displayed as static images.

### **How to add Gantt chart to your application**

To add support for Gantt chart in your application you need to do the following:

1. Add some predefined business objects to your application that represent Gantt tasks, dependencies and allocated resources
2. Add business rules and process to create and process instances of these objects
3. Add a query that finds tasks and displays them and related dependencies as a Gantt chart

To add predefined business objects to your application select the Business Objects node in the tree, right click and select the "Add Gantt Task Object" command from the popup menu. Before doing this, make sure that you define your own object that represents a "resource" that will be allocated to a task (even if you don't use resources in your Gantt). It can be any object that has an attribute called "Name". After you select the command Aware IM will ask you to provide the name of the "resource" object and then it will create predefined objects called GanttTask and GtDependency with all the required attributes (Note that GtDependency has references to the tasks that it "connects").

After this you need to define rules and processes that will create and manage tasks and related dependencies.

Finally you need to define a query to find task objects and display them as a Gantt chart. When a query looks for task objects Aware IM will automatically enable the Gantt option in the Display options of the query. You can then specify properties of the Gantt chart that you want to show. The Gantt chart will automatically include all dependencies related to the tasks.

For an example of the Gantt chart implementation take a look at the Project Management sample application. See also a [video tutorial](#) about Gantt charts.

### **How to use charts in reports**

You can either use "chart report elements" or use images of online charts. For the description of the chart report elements please refer to the documentation of reports in the User Guide. To use images of online charts in your reports or documents do the following:

- Define a query to display your chart (see Chart presentation of query results in the User Guide)
- When defining chart properties of the query specify a unique id of the chart (can be any string)

- Define an attribute of the Picture type in some object to store the image of the chart
- Create a document template that would use this attribute in a tag expression, for example <<MyObject.ChartImage>>
- Write a process that will save the image of the chart in this attribute using the SAVE SCREEN action (see the User Guide for details) and display the document template, for example:

```
SAVE SCREEN 'my chart id' TO MyObject.ChartImage  
DISPLAY DOCUMENT 'My template using chart image'
```

### **What is the difference between processes and business rules?**

Business rules encapsulate business logic of the system, i.e. they encapsulate any rules that are not related to the user interface (or any other external interface to the system). Processes, on the other hand, provide a bridge between business rules and the user interface (or any other external interface to the system). Usually processes just activate business rules in response to some external request and deliver the end result to the requestor. Therefore processes should not include business logic in their rules and conversely, business rules should not invoke actions that interact with a user (such as ENTER NEW, EDIT, PICK FROM and others). See also the “Business Rules as Carriers of Business Logic”, “Processes as Links Between User Interface and Business Logic”, “Configuration Guidelines” sections in the Aware IM User Guide. See also [How to go about system configuration](#).

### **How to define a rule**

Most business rules are evaluated when a particular business object is created or modified and therefore most rules are attached to business objects. Some rules may also be attached to notifications (they are evaluated when a notification is created or received) or to timer events (scheduling rules). A rule consists of a condition (which is optional) and actions. To define a rule you need to specify these conditions and actions using the Rule Language. To make this task simpler **Aware IM** provides the Context Assistant, which prompts you for the right actions and conditional expressions to use when defining a rule. You can also use the information provided in the “Rule Language Reference” section in the Aware IM User Guide or in the “Aware IM Rule Language Reference” document.

### **How to find a rule**

An application that you configure may have many rules. If you want to find a particular rule in the Configuration Tool you can use “Search Rules” functionality – see the “Searching Rules” section in the Aware IM User Guide. You can also use various filtering options available in the Rule Collection Editor. The Rule Collection Editor may display all rules in the application or only rules attached to



a particular business object or notification – see the “Working with Rule Collection Editor” section in the Aware IM User Guide.

### **How to deal with rules that need to be evaluated in a strict order**

Quite often you are dealing with the situation when rules need to be evaluated only after the action of the previous rule has been executed, i.e you seem to be dealing with the ordered sequence of rules. However, in **Aware IM** most rule sequences are un-ordered (except processes that have “Maintain Order” property checked), i.e. **Aware IM** does not guarantee the order in which rules will be evaluated (see the “Rule Evaluation” section in the Aware IM User Guide).

One way around this is to use an ordered process, however, this approach is not recommended as the general solution, as the process should not contain any business logic (see [What is the difference between processes and business rules](#)). The better solution is to get rid of the explicit order and break up an ordered rule sequence into a number of independent rules that get triggered when the value of some attribute or list changes. One way to do this is to introduce the “State” attribute to the object. The “first” rule may set the state to some value and the “second” rule can be made dependent on this value. The “second” rule can then set the state to some other value which the “third” rule will check and so on. Note that although we are using the terms “second”, “third” etc, each rule is independent of other rules as it only depends on whether the conditions that it checks are true or false (it does not care who sets the value of the state attribute and when - as long as the state has a particular value the rule will be evaluated). State attribute is often a good solution to the problem, however, any other trigger will do (for example, you can check whether an object has been added to a list) – as long as one rule sets the trigger and another rule checks it, we can express any ordered sequence as an un-ordered collection of independent rules.

### **How to make sure that action is executed only once**

Most rules in **Aware IM** are un-ordered, i.e. **Aware IM** does not guarantee the order in which the rules will be evaluated. Consequently, the same rule may be submitted for evaluation several times and so its action may be executed several times as well (see the “Rule Evaluation” section in the Aware IM User Guide). For actions that modify attribute values this is not a problem, provided that the last execution of the action sets the attribute to the correct value. **Aware IM** guarantees that this will be the case provided that rules are consistent and independent of one another.

What about rules that perform a different action, for example report errors or request services? Will they be executed several times as well? The answer is no, because **Aware IM** assigns different priorities to rules invoking such actions to make sure that they are evaluated after the rules that modify attributes (see the

“Rule Priorities” section in the Aware IM User Guide). Even so, it is probably not a very good idea to execute certain actions from rules attached to business objects without taking special precautions, as the rules will be evaluated every time the object is changed. For example, let us consider the following rule:

```
If Account.State = 'Open' Then CREATE AccountOpeningLetter  
WITH AccountOpeningLetter.Addressee = Account.Holder
```

The way the rule is written the letter to account holder will be created every time the Account object is changed (provided that it is in the “Open” state) – even if changes have nothing to do with the state of the account. This is not what we want. What we really want to do is to create a letter only when the account *becomes open*. Therefore we have to write our rule as follows:

```
If Account.State WAS CHANGED TO 'Open' Then CREATE  
AccountOpeningLetter WITH AccountOpeningLetter.Addressee =  
Account.Holder
```

In certain cases it may even be necessary to introduce a state or some other attribute to check whether an action has been performed, for example:

```
If Policy.ExpiryDate<=CURRENT_DATE AND  
Policy.LetterCreated='No' Then CREATE ExpiryLetter WITH  
ExpiryLetter.Addressee = Policy.Holder
```

Note that it is *not* necessary to take any precautions when sending e-mails – **Aware IM** will send all e-mails only after all rules have been evaluated (more precisely, e-mails just like any other notifications are sent when the current transaction is committed – see the “Rules and Transactions” section in the Aware IM User Guide).

### **How to perform an action on multiple business objects**

Aware IM Rule Language is the language of rules only – it does not include variables or loops as programming languages do. Consequently, it is not possible to iterate over business objects. How can then an action be performed on several business objects? The answer is that any action is automatically performed on all objects that are in the current *context*. The context usually contains objects found by a query, so the action is performed on all found objects. For example,

```
FIND Account WHERE Account.Balance < 100  
Account.State='CLOSED'
```

The second action that changes the state of the account to “CLOSED” will be performed on all objects found by the `FIND` action and so all accounts with balances less than 100 will be closed.

### **How to check whether rules are executing correctly**

Execution of rules can be checked using the Execution Log and Log Viewer. See the “Execution Log” section in the Aware IM User Guide.

### **How to design a report**

To design a report you have to configure a document template of the Report type and define its layout using the Report/Presentation Designer. See the “Adding/Editing Document Templates” section in the Aware IM User Guide for details.

### **How to produce and print out a report**

For the user to be able to produce and print out a report the report layout and the corresponding operation have to be configured. To design a report you have to configure a document template of the Report type (report template) and define its layout using the Report/Presentation Designer (see the “Adding/Editing Document Templates” section in the Aware IM User Guide). To get **Aware IM** to produce the report so that the user can see it and print it out you can do either of the following:

- Define a menu item of the Create Document type (see the “Setting Menu Item Properties” section in the Aware IM User Guide). **Aware IM** includes the default menu item of the Create Document type that allows the user to select the report she wants to create among all the report templates defined. Alternatively you can configure your own menu item that will generate the report from your specific report template. When a user selects this menu item in the Operation Mode **Aware IM** will generate the report and show it on the screen as PDF document (then it can be printed from the browser). A report created in this manner is not stored in the system and have to be re-created each time.
- Define an attribute of the Document type in some business object and set its initial value to the name of your report template (see the “Attributes of Document Type” section in the Aware IM User Guide). Whenever the instance of the business object is created the report will be generated and stored permanently in PDF format as the value of the attribute. The user can bring up the form of the business object and press the View Document button to get the report to be displayed on the screen (then it can be printed from the browser). A report created in this manner gets permanently stored in the system together with the instance of the business object.
- Define an operation of the “Create Document” type on a form of a business object – see the “Adding/Editing Form Operations” section in the Aware IM User Guide.
- A report may be generated as part of the process that uses `DISPLAY DOCUMENT` action. When the action is executed the report is displayed on the screen as PDF document. Note that the action can generate the report from

the template as in the first option or display the document already stored as the value of the attribute of the Document type as in the second option.

### **How to refer to database fields in a report**

When you design a report you can refer to attributes of business objects using tag elements – see the “Document Generation” section in the Aware IM User Guide for details.

### **How to perform calculations in a report**

When you design a report you can define calculations to be performed inside tag elements. You can use any expressions allowed by the Rule Language. The common expression is, for example, to calculate the sum total of the attribute:

```
<<SUM Account.Balance>>
```

See the “Document Generation” section in the Aware IM User Guide for more details.

### **How to group records in a report**

Sometimes you may want to group the data shown in a report by some criteria and print out a special header and/or footer for each group. To do this you need to configure the following:

1. The query that provides the data for the report must sort the data accordingly.
2. The layout of the report must include group bands. Each group band must have a group condition defined.

See the “Report Generation” and “Setting Band Properties” sections in the Aware IM User Guide for more details.

### **How to include related records in a report**

You can show values of attributes of business objects related to a particular business object on a report. For example, if a PurchaseOrder object is related to an OrderLineItem object via a multiple relationship you can show which order line items are related to a particular purchase order on the report.

To include values of attributes of the related object in the report you have to define a sub-report element in the report. The sub-report element must refer to the report showing the instances of the related object and the query of the sub-report element must find the related objects. For example:

```
FIND OrderLineItem WHERE OrderLineItem IN  
PurchaseOrder.MyLineItems
```

See also the “Editing Sub-report/Sub-presentation Element” section in the Aware IM User Guide.

### **How to include an element in a report conditionally**

Sometimes it may be necessary to include a particular element (such as text or image) in the report only under certain conditions (for example, you may show sensitive data to one category of users but not others). In this case you can define a conditional element in the layout of your report and specify the conditions of printing. See the “Reports” and “Editing Conditional Element” sections in the Aware IM User Guide for details.

### **How to print an element in different colors depending on some conditions**

Sometimes the same element may be shown with different properties (for example, in different colors) depending on some conditions – for example you can show the account balance in red if it is less than, say, 100 dollars and in green otherwise. This can be achieved by defining a conditional element in the layout of the report. The element will print the text in different colors depending on the conditions - see the “Reports” and “Editing Conditional Element” sections in the Aware IM User Guide for more details.

### **How to enumerate elements in a report**

Quite often you need to print out either the total number of data records in the report (or page or group) or the sequence number of a data record in the report (or page or group). This can be achieved by using the `ELEMENT_COUNT` or `ELEMENT_COUNTER` functions. If you use `ELEMENT_COUNT` function inside a tag element the report will print out the total number of data records in the report (or page or group). If you use `ELEMENT_COUNTER` function inside the tag element the report will print out the sequence number of the data record in the report (or page or group). See the “Editing Tag Element” section in the Aware IM User Guide for more details. You can also use other report specific functions such as `PAGE_NUMBER` and `NUMBER_OF_PAGES`. The full list of such functions is provided in the “Miscellaneous Functions” section in the Aware IM User Guide.

### **How to re-use a report**

If you want to use the layout of a report in a different report you can export the layout of the first report and then import it into the second report. See the “Adding/Editing Document Templates” section in the Aware IM User Guide for details on how to export and import the report layout.

## **How to store documents and files**

In **Aware IM** documents and files are stored in attributes of the Document type. See the “Attributes of Document Type” and “Document Management” sections in the Aware IM User Guide for more details.

## **How to store pictures**

In **Aware IM** picture are stored in attributes of the Picture type. See the “Attributes of Picture Type” section in the Aware IM User Guide for more details.

## **How to do "document merge"**

**Aware IM** can perform “document merge” of Microsoft Word documents, Microsoft Excel documents, plain text documents and HTML documents. If you want to provide the “document merge” functionality in your system you have to configure document templates of any of the above mentioned types (see the “Adding/Editing Document Templates” section in the Aware IM User Guide). Alternatively you can use document templates defined by the end user (Developer Edition only)

The templates must include tags that will be “merged” with real data when the final document is produced – see the “Document Generation” section in the Aware IM User Guide for details on how it is done. To get **Aware IM** to produce the “merged document” so that the user can see it and print it out you can do either of the following:

- Define a menu item of the Create Document type (see the “Setting Menu Item Properties” section in the Aware IM User Guide). **Aware IM** includes the default menu item of the Create Document type that allows the user to select the document she wants to create among all the document templates defined. Alternatively you can configure your own menu item that will generate the document from your specific document template. When a user selects this menu item in the Operation Mode **Aware IM** will generate the “merged document” and show it on the screen (then it can be printed from the browser). A document created in this manner is not stored in the system and have to be re-created each time.
- Define an attribute of the Document type in some business object and set its initial value to the name of your document template (see the “Attributes of Document Type” section in the Aware IM User Guide). Whenever the instance of the business object is created the “merged document” will be generated and stored permanently as the value of the attribute. The user can bring up the form of the business object and press the View Document button to get the document to be displayed on the screen (then it can be printed from the browser). A document created in this manner gets permanently stored in the system together with the instance of the business object.

- A document may be generated as part of the process that uses `DISPLAY DOCUMENT` action. When the action is executed the document is displayed on the screen. Note that the action can generate the document from the template as in the first option or display the document already stored as the value of the attribute of the Document type as in the second option.
- Define a form operation of the Create Document type that can be invoked from the form of the business object. The instance of the object being edited will be used as the data source for the “merged document”. See the “Adding/Editing Form Operations” section in the Aware IM User Guide for more details.

The data that **Aware IM** will use for the document may come from several sources:

- The user may explicitly specify a query to be used as part of the Create Document menu command
- The user may explicitly select certain instances of objects to be used as the data source when viewing the results of some query and then run the Create Document operation
- You can specify the query in your process as the FIND action that will put the desired data into the context.
- You can specify the query in the `DISPLAY DOCUMENT` action (“USING” clause)
- You can specify the query when configuring the document template

The preferred approach is to use the `DISPLAY DOCUMENT` action in a process. The action may use the query as the data source. If you are displaying the report **Aware IM** will just display it on screen. For other types of documents **Aware IM** will generate as many documents as there are instances returned by the query. You can get **Aware IM** to display the results on screen and let the user select specific instances to print out. Or you can use the `MERGE INTO ONE` clause in the action in which case Aware IM will merge all documents into a single document and show it immediately to the user who can then print it out.

### **How to let end users define their own reports, document templates and queries**

This is described in the “User Defined Documents and Reports” and “Building Query” sections in the Aware IM User Guide.

### **How to implement multiple file upload for documents**

Aware IM does not support “multiple allowed” for attributes of the Document type, however, there is a way for you to implement “Multiple file upload” functionality when the user selects several documents at once and uploads them all to the system. In order to implement this functionality you need the following:

- Define a business object representing your document. This business object should have at least one attribute of the Document type. It can have other attributes as well, such as Description of the document, type etc
- Make sure that this object can be created automatically, i.e. if it has some mandatory attributes, the initial values for these attributes are provided. This is because Aware IM will be creating instances of this object behind the scenes and if some required values are not initialised properly by rules Aware IM will not be able to create instances of the object
- Define a multiple reference to this object in the business object that will store the documents. In other words, rather than defining a Document attribute in this object you define a multiple reference to the object that includes this Document. For example, if you want to define attachments to your Order object, first define an `OrderAttachment` object with the Attachment Document attribute and then define multiple reference attribute Attachments (of the OrderAttachment type) to the Order object
- Go to the Presentation option of the multiple reference attribute (Attachments in the example above) and go to the Panel Operations. Then add the Add New operation.
- Select “Use multi-file upload for document attribute” as the parameter of the operation and specify the name of the Document attribute. At run time the user will be able to select multiple files to upload and for each file Aware IM will create an instance of the Document wrapper object (`ObjectAttachment`), upload the file into the Document attribute of this object and attach the instance of this object as a reference (to the Order object)

Starting from version 5.7 it is now possible to implement multiple file upload as a process. To do this you need to write a process that assumes that the Context contains a predefined business object called `FileItem`. This object has one attribute of the Document type called `Doc`. The process receives multiple instances of this object and it can do something with the documents stored in the `Doc` attribute – for example, copy them to some other business object, for example:

```
CREATE Photo FOR EACH FileItem WITH Photo.Picture =  
FileItem.Doc
```

### **How to let users edit documents**

There are two ways you can let users edit documents that they store in the attribute of the Document type.

The default approach is to let users download the document, then edit the document outside of Aware IM using the appropriate editor and then let them upload the modified document back into the Document attribute of the business object. When you define an attribute of the Document type Aware IM



automatically provides the buttons to upload the document to Aware IM and download it from Aware IM, so this approach is quite straightforward. The disadvantage of this approach is that the user has to perform quite a few steps to modify the document – download, edit and upload back.

An alternative approach is to use Zoho or Office Online document editors to edit/view the document in-place. Zoho (<http://www.zoho.com>) provides powerful web-based editors to edit many types of documents – MS Word, MS Excel, MS Power Point, Open Office, RTF, HTML etc. The user edits the document using the browser and when he presses the Save button the modified document is automatically saved in Aware IM.

Office Online allows editing Microsoft Office documents only. Just like Zoho the user edits the document using the browser and the changes are automatically saved in Aware IM. To edit documents using Office Online you have to make sure that your server where Aware IM is running meets Microsoft requirements –

- It has to be added to the Microsoft “WOPI allow list” – please contact Microsoft for this
- Tomcat has to be setup to run on either port 80 (testing mode only) or port 443 (both testing and production modes)

By default Aware IM runs Office Online in the testing mode. To enable the production mode create the file called “oo.props” that contains the following property:

```
DiscoveryUrl=https://onenote.officeapps.live.com/hosting/discovery
```

Put this file in the AwareIM/Tomcat/webapps/AwareIM directory.

See also this article for more details about putting Office Online in the production mode: [https://wopi.readthedocs.io/en/latest/build\\_test\\_ship/shipping.html](https://wopi.readthedocs.io/en/latest/build_test_ship/shipping.html)

To enable document editing feature the configurator has to do the following:

1. For Zoho integration make a decision whether their system will use their users’ Zoho account or their own Zoho account. If they want to use their own account they need to register this account with Zoho and then go to the following URL <https://zapi.zoho.com> to create the API key.
2. Then they need to create an attribute of the Document type as usual, go to the Presentation options of this attribute and tick the “Zoho editing” or “Office Online editing” checkboxes
3. For Zoho integration click the Settings button next to the checkbox and provide the Zoho API key
4. At runtime Aware IM will automatically display the editing button next to the attribute. When the user clicks on this button the corresponding Zoho or Office Online editor will be invoked.
5. If configurators want their users to use their own Zoho account, they will need to define a special attribute in the SystemSettings business object

that will hold the Zoho API key. At step 4 they need to provide the name of this attribute, rather than the API key itself. Before the system is used by customers their system administrator will need to perform step 2 to register Zoho API key and set its value into the appropriate attribute of the SystemSettings object.

**IMPORTANT:** If you want Zoho to edit documents inline the URL that your users use to start the application **MUST** use the full server name – localhost or raw IP address will not work. Also, Aware IM web server should be set up to run on port 80 (or 443 if using SSL) For example:

<http://myserver.com/AwareIM/logon.html>  
(this will allow editing)

<http://localhost/AwareIM/logon.html>  
(this will not allow editing – just viewing)

<http://205.46.67.78/AwareIM/logon.html>  
(this will not allow editing – just viewing)

<http://myserver.com:8080/AwareIM/logon.html>  
(this will not allow editing – just viewing)

Note also that Zoho editing will not work in the testing mode either.

### **How to work with embedded spreadsheets**

Embedded spreadsheets can be used to display data as an Excel table or to get Excel to perform calculations that can then be written into attributes of business objects and stored in the database. This can be especially useful if you already have existing Excel spreadsheets that you want to re-use to perform some calculations.

To use embedded spreadsheets in Aware IM you need to do the following:

- Define an attribute of the Document type in some object
- Open this attribute for editing, select the “Allowed Extensions” property and then choose “Embedded Spreadsheet” from the drop down list of allowed extensions.
- If you will be storing results of spreadsheet calculations in the database define attributes that will store these calculations (the attributes must belong to the same object that owns the Document attribute)

When the system shows the form of this object in the browser Aware IM will generate a button that will allow users to create and edit spreadsheets. When the user presses this button Aware IM will display an Excel-like spreadsheet. You can create your own spreadsheet or use an existing one by importing an Excel

file or using Copy and Paste to paste cells from Excel into the Aware IM spreadsheet.

If you want Aware IM to populate an existing spreadsheet with data from the database you need to specify the contents of the cell as “tag expression” that refers to objects in Context, for example <<Customer.Name>>. The Context will contain the instance of the object that owns the Document attribute with the spreadsheet. When the user presses the “Read Data” button at the top of the spreadsheet Aware IM will replace contents of the cell that have tag expressions with data from the database.

Once the data has been populated Excel will obviously re-calculate everything based on the values read from the database. You can then save results of the calculation in attributes of the object(s) in Context. To do this you need to do the following:

- Click on the “Mapping” button at the top of the spreadsheet to define cells that will be written into attributes of the object(s) in Context. You need to define which cell to write and then provide a tag expression that defines where this cell will be written into. Note that the mapping is saved with the spreadsheet, so you only need to define mapping once
- Click on the “Save Data” button to get Aware IM to save the cells defined in “mapping” into the database

### **How to set up system users**

If your application is a multi-user application and you want to control access of these users to the application then you have to do the following:

1. When configuring the application make sure that it has the access level that users will be working at (see the “Access Level” section in the Aware IM User Guide). It can be the predefined Guest access level or, more likely, you will configure the special access level for this. Do not let users work at the Administrator access level, as this predefined access level should only be used by the system administrator(s).
2. Make sure that the default visual perspective for the Administrator access level includes the menu item of the Create Object type that allows creating instances of the predefined RegularUser business object (**Aware IM** includes such an item by default) – see the “Setting Menu Item Properties” section in the Aware IM User Guide and [How to create records](#).
3. Make sure that the default visual perspective for the user access level does not allow creating the RegularUser business object, but does include the menu item of the Change Login Details type that allows users to change their initial login name and password.
4. Once the application has been configured and published the system administrator has to log in the operation mode using the administrator password (see the “Full User Login, Interactive” section in the Aware IM User

Guide). The administrator password is the same password as the one used to log into the Configuration Tool (see also [How to set/change password for the Configuration Tool](#)).

5. Once inside the Operation Mode the system administrator has to create instances of the RegularUser object – one instance per user. When creating instances of the RegularUser object the system administrator has to specify values for the following attributes - login name, password and access level. Thus the new user of the system will be assigned the initial login name, the initial password and a particular access level.
6. Users can now log into the system using the assigned login names and password and then change these initial values using the menu item of the Change Login Details type. The users should not be able to change the value of access level assigned by the system administrator.

If you want to assign additional attributes to the system users (for example, their personal details) you can add attributes to the RegularUser object or create a different business object representing the user. In the latter case you must add this business object to the predefined SystemUsers business object group (see the “Adding/Editing Business Object Groups” section in the Aware IM User Guide). This will make sure that the new business object will have certain predefined attributes required by the system such as the login name, password and access level. All other steps should be the same – the only difference is that instead of creating instances of the RegularUser object you will be creating instances of your custom user object.

### **How to log into the system**

Most users have to log into the application in the operation mode before they can start using it (this includes system administrators as well). The only exception is “guest” users that can work with the system without having to log in (“guest users”, however, are heavily restricted in what they can do with the system). The standard way of logging into the system, which should be used by most users, is described in the “Simple User Login, Default” section in the Aware IM User Guide. System administrators can also use other entry points – all different ways of logging into the system are described in the “Login” section in the Aware IM User Guide.

### **How to set/change user password**

The initial password for users of the application can be set up by the system administrator – see [How to set up system users](#). Once users log in using the credentials created by the system administrator, they can change these credentials provided that a menu item of the Change Login Details type has been configured for them.

## **How to set up users of the Configuration Tool**

**Aware IM** creates one default user of the Configuration Tool during installation. This user has the predefined login name “admin” and the predefined password “password”. Thus the configurator can log into the Configuration Tool using the credentials of the default user. The default password should be changed immediately once inside the Configuration Tool using the “Change Administrator Password” command.

If you want to create additional users of the Configuration Tool you have to do the following:

1. Log into the system in the Operation Mode using the credentials of the default user (you do not have to configure anything for this – use the default configuration provided by **Aware IM** after the installation). You can log in directly from the Configuration Tool – see the “Logging into the Operation Mode” section in the Aware IM User Guide.
2. Use Create Object menu item in the Standard folder to create instances of the RegularUser business object – one instance per configurator. When creating the instances you have to set the values for the following attributes – login name, password and access level. Set the login name and assign initial value to the password attribute (the password can be changed later by the configurators). Make sure that access level is set to “Administrator”.
3. The configurators can now log into the Configuration Tool using the specified credentials.

## **How to set/change password for the Configuration Tool**

**Aware IM** creates one default user of the Configuration Tool during installation. This user has the predefined login name “admin” and the predefined password “password”. Thus the configurator can log into the Configuration Tool using the credentials of the default user. The default password should be changed immediately once inside the Configuration Tool using the “Change Administrator Password” command.

## **How to allow non-registered users access to the system**

If you make your application widely available over the Internet you may want to allow non-registered users to access the application too, i.e. allow access to those users who do not have to login. Such users can enter the system using the URL described in the “Guest Entry” section in the Aware IM User Guide. All users who enter the system via this URL are automatically assigned “Guest” access level (see the “Predefined Access Levels” section in the Aware IM User Guide). By default this access level is very heavily restricted – you may want to relax these restrictions a little bit to allow guest users certain operations with the system. You may also want to include the menu items of the Login and Register User types into the visual perspective for the Guest access level to allow “guest

users” to login or self-register – see the “Setting Menu Item Properties” section in the Aware IM User Guide.

### **How to allow non-registered users to self-register**

Make sure that you include menu item of the Register User type into the visual perspective for the Guest access level – see [How to allow non-registered users to access the system](#). You can also define a special form in the business object that represents a user, which will be used for this occasion – see the “Defining Business Object Forms” section in the Aware IM User Guide, [How to set up system users](#).

### **How to refer to the current user in rules**

You can use LoggedIn prefix – see the “Instance Prefixes” section in the Aware IM User Guide. For example,

```
If LoggedInLibraryMember.LoginName ='john smith' Then ...
```

### **How to use Active Directory / LDAP for login**

This is explained in the “Using LDAP/Active Directory for Login” section in the User Guide

### **How to implement “forgotten password” functionality**

Your application may need to provide the “forgotten password” functionality. Usually this functionality asks the user one or more security questions and then compares their answers with predefined values. Then the system either tells/e-mails the password to the user or asks them to provide the new password. It is entirely up to you how you want to handle this. In order to define the forgotten password functionality you need to do the following:

1. Define a process that would handle this functionality – ask the user the security questions, compare answers, set new password etc
2. Tick the “Handles forgotten password” functionality checkbox when defining the process
3. Use the login form that includes the link that invokes the “forgotten process” functionality. The name of the predefined form is “login2.html”. Alternatively you can provide your own login form and include the link to the following URL:

```
logonFP.aw?domain=YourBusinessSpaceName
```

The link will automatically start the process that is marked as “handles forgotten password” functionality.

### **How to set up Login via Facebook, Twitter and Google**

If you want your users to be able to login to Aware IM automatically using their existing Facebook, Twitter or Google accounts you need to do the following:

1. Register your application as Facebook, Twitter or Google developer (see [How to register Aware IM application in Facebook](#), [How to register Aware IM application in Twitter](#), [How to register Aware IM application in Google](#) sections for more details)
2. In the Configuration Tool double click on the business space version you want to change and then select the `Login Options` property from the list of properties.
3. Tick the corresponding checkbox to allow login via a particular account, for example `Allow Login via Facebook`
4. Click on the `Settings` button next to the tickbox.
5. Enter the application id and secret obtained from step 1.
6. Alternatively go to the `SystemSettings` business object and add two attributes to this object to hold the values of the application ID and secret (that is, if you want system administrators to be able to do this at run-time) Then specify these attributes in the `Settings` dialog.
7. It is important to understand that when the user logs in via Facebook, Twitter or Google Aware IM will find the instance of the user object that corresponds to the current user. Therefore, you need to indicate which type of the user object Aware IM should look for, i.e. you need to indicate the member of the `SystemUser` group in the user drop down.
8. You should then select the attributes that Facebook, Twitter or Google provides that should map to the attributes of the selected user object. For example, you can map the `id` attribute provided by Facebook to the `LoginName` attribute of the user. Every time a user with this Facebook id logs in Aware IM will find the corresponding user object with the value of `LoginName` equal to this id.
9. You should also tick the “Allow creating the object automatically...” checkbox if you want Aware IM to automatically create a user instance if this instance has not been found. Note that the user object should not have any mandatory attributes that are not provided by Facebook, Twitter or Google, otherwise Aware IM will not be able to create the instance automatically
10. The user can then use a special URL to login. See see [How to register Aware IM application in Facebook](#), [How to register Aware IM application in Twitter](#), [How to register Aware IM application in Google](#) for more details

### **How to store “session” values – values that exist only during user session**

Sometimes it may be necessary to store some values specific to the session of a particular user – for example, remember current selection of the user. You may want to use such values in rules, but you don't want them to be persisted in the database, because they do not make sense when the user logs out.

Aware IM supports a concept of the “session object”. The instance of this object is created once per user – when the user logs in. This instance then becomes

automatically available in Context of business rules (you don't need to find it), for example `IF MySessionObject.CurrentSelection = 'abc' THEN ...`

To define such an object you can create any object with any attributes as required and then set its Persistence property to "Not Persisted". You should then tick the "Stores session values" checkbox to indicate that this object is the "session" object. Note that you can set only one business object as "session" object.

### **How to configure the system to send e-mails**

To configure the application to send e-mails you need to do the following:

1. Define an intelligent business object representing the addressee of the e-mail with the communication channel of the "e-mail" type (see the "Defining Intelligent Business Objects" section in the Aware IM User Guide).
2. Define a notification representing a particular e-mail (see "Sending Outgoing E-mail" section in the Aware IM User Guide).
3. Define a business rule that would use the `SEND` action to send e-mail under particular circumstances – see the description of the `SEND` Action in the Aware IM Rule Language Reference.

See also the "Outgoing E-mail" section in the Aware IM User Guide.

### **How to send e-mail with attachments**

To send e-mails with attachments you need to configure a notification representing a particular e-mail (see the "Sending Outgoing E-mail" section in the Aware IM User Guide) and define attributes of the Document type that will hold the attachments – one attribute per attachment (see the "Attributes of Document Type" section in the Aware IM User Guide). When you send e-mail using the `SEND` action you need to initialize these attributes with the particular documents (these can be generated reports, "merged documents" or ordinary files). **Aware IM** will automatically convert these documents into e-mail attachments. For example,

```
CREATE EmailToCustomer WITH
EmailToCustomer.Attachment1='AnnualReport'
SEND EmailToCustomer TO Customer
```

In this example `EmailToCustomer` is a notification representing the e-mail (the notification must have the `Subject` and `Message` attributes initialized to some text). This notification has the attribute `Attachment1` of the Document type. "AnnualReport" is the name of the document template of the Report type (see the "Reports" section in the Aware IM User Guide). When initializing the `Attachment1` attribute **Aware IM** will generate the report, attach it to the attribute and convert it to the e-mail attachment when the e-mail is sent. Another example:



```
CREATE EmailToCustomer WITH
EmailToCustomer.Attachment1='AnnualReport'
IMPORT DOCUMENT EmailToCustomer.Attachment2 FROM
'c:\MyDocuments\myfile.bin'
SEND EmailToCustomer TO Customer
```

In this example EmailToCustomer is assumed to have the second attribute of the Document type called Attachment2. After the notification has been created the value of this attribute is imported from file. When the e-mail is sent the file is converted to the e-mail attachment.

### **How to configure the system to receive e-mails**

This is described in detail in the “Incoming E-mail” and “Setting Options for Incoming E-mail Handling” sections in the Aware IM User Guide.

### **How to register (save) incoming e-mails in the system**

You can ask **Aware IM** to register all incoming e-mails when you configure the properties of the incoming e-mail handling – see the “Setting Options for Incoming E-mail Handling” section in the Aware IM User Guide.

### **How to process e-mail based on its contents**

In order to process the received e-mails you need to define business rules attached to the event of receiving an e-mail notification – see the “Incoming E-mail” section in the Aware IM User Guide. The rules may check the subject and/or body of the e-mail (or maybe who it is from and when it was sent) and perform the appropriate actions. For example,

```
If IncomingEmail.Subject = 'Registration' AND
IncomingEmail.From = 'John Smith' Then ...
```

If e-mails contain complex information that you want to analyse then this information may be encoded into the body of the message in a special format (you have to mandate that the senders of such e-mails encode the bodies of their e-mails in this format). The format is described in the “SET Action” section of the Rule Language Reference. Your rules can then decode the information contained in the body of the e-mail using the SET action. For example, if the body of the e-mail has the following string:

```
#Name#John Smith#State#NEW#Balance#100.0
```

Then the following rule:

```
SET Account FROM IncomingEmail.Message
```

will set the Name, State and Balance attributes of the Account objects to the values provided in the body of the e-mail ('John Smith', 'NEW' and 100.0). You can define other rules that will check these values and perform the appropriate actions if required.

### **How to do "mail-merge"**

To perform "mail-merge" of your outgoing e-mails do the following:

1. Define a notification representing the e-mail – see the "Sending Outgoing E-mail" section in the Aware IM User Guide, [How to configure the system to send outgoing e-mails](#)
2. Provide the initial values to the Message and Subject attributes of the e-mail notification to include the body and subject of your e-mail respectively (see the "Common Properties" section in the Aware IM User Guide). Use tags inside the initialization text for those parts of the body and/or subject that will be "mail merged". **Aware IM** will replace the tags with the appropriate values when the e-mail is sent. Below is an example of the initialization text for the Message attribute:

```
Dear <<Policy.Holder.Name>>,  
Your policy is about to expire on <<Policy.ExpiryDate>>
```

### **How to send e-mails in HTML format**

To send e-mails in the HTML format first define the text of the e-mail as HTML document template (see the "Adding/Editing Document Templates section in the Aware IM User Guide). After that add an email notification as described in [How to configure the system to send e-mails](#). This notification will have the predefined HTMLMessage attribute of the Document type. Set initial value of this attribute to the name of the HTML document template. Then define a rule to send this e-mail as usual.

### **How to give different authority to different users**

Most multi-user systems require that different categories of users have access to different features of the system. In **Aware IM** this is done by assigning different access levels to different users – see the "Access Level" section in the Aware IM User Guide for details.

### **How to limit access to database tables**

In **Aware IM** you are not dealing with database tables directly – instead you can limit access to business objects – see the "Setting Access to Business Objects" section in the Aware IM User Guide for details. You can also limit access to

business objects based on some conditions – see [How to define access based on some conditions](#).

### **How to limit access to records**

In *Aware IM* you are not dealing with database records directly – instead you can limit access to attributes of business objects – see the “Setting Access to Business Objects” section in the Aware IM User Guide for details. You can also limit access to attributes based on some conditions – see [How to define access based on some conditions](#).

### **How to limit access to operations**

Most operations are invoked from forms - see [How to use forms for different users](#). You can also limit access to processes and services – see the “Setting Access to Processes” and “Setting Access to Services” sections in the Aware IM User Guide.

You can also define an “applicability condition” when defining an operation, for example:

```
Member.Status = 'Active'
```

The operation will only be shown for those members whose status is Active.

### **How to define access based on some conditions**

If you want to limit access to business objects or its attributes based on some conditions you have to specify business rules that use the `PROTECT` action. See the description of the `PROTECT` action in the Rule Language Reference for details.

### **How to get end users to define their own access restrictions**

Sometimes it may be necessary for end users (for example, system administrators) to override access restrictions supplied by developers. This could be the case if developers sell their application to many different customers each of them having their own unique access level requirements.

This can be done as part of the “User Defined Processes/Workflows” module. A developer needs to include a menu command of the “Configuration of User Defined Processes type”. This menu command brings up the user defined processes configuration module that allows among other things to define access restrictions to business objects and attributes.

### **How to implement multi-tenancy**

It is often required that the same application is hosted for different “tenants”. A tenant logs into the system and only sees data that belongs to him, but not other tenants.

There are two approaches in Aware IM on how this can be implemented:

1. Each tenant has a separate business space allocated to the tenant and the same application is loaded into each business space
2. All tenants are managed by the same business space and the application itself is defined in such a way that each tenant’s data is protected from another tenant

There are pros and cons in both approaches, so the approach you should take depends on your specific requirements.

The advantage of the first approach is that each tenant is completely independent from each other because each business space is totally separate from another one. Moreover, you can allocate a separate database to each business space, so that the data of each tenant is stored in its own database. The application is simpler because there is nothing in it that refers to tenants. Data backup can be done on a tenant-by-tenant basis and a data problem in one tenant doesn’t affect another tenants.

The disadvantage of the first approach is that if there are thousands of tenants, existence of thousands of business spaces can significantly impact the performance of the server (however, you can add more servers to alleviate the problem). Another disadvantage is that if your applications needs functionality that works across different tenants – for example, if you want to collect statistics about your tenants, look at all registered tenants etc, it may be hard to do if all your tenants are independent applications.

Advantages and disadvantages of the second approach are reverse of that of the first one. The second approach may not depend as much on the larger number of tenants as the first approach and it’s also quite easy to provide functionality that looks across different tenants. But the application itself becomes more complicated and it is a lot more difficult to provide backup/restore on a tenant-by-tenant basis (but not impossible).

The following paragraph describes what needs to be done in the application to implement the second approach. This is what you need to do:

1. Define an object representing a Tenant
2. For all objects that belong to the tenant add a reference to the Tenant that it belongs to, for example Account.Tenant
3. The user object must have a reference to Tenant that it operates on behalf of - LoggedInSystemUser.Tenant
4. For each object that belongs to some tenant add rules to initialise tenant from

the tenant of the logged in user that created the object, for example:

```
IF Account.Tenant IS UNDEFINED Then  
Account.Tenant = LoggedInSystemUser.Tenant
```

5. Finally add protection rules to the object that belong to the tenant so that other tenants do not see it, for example:

```
IF Account.Tenant <> LoggedInSystemUser.Tenant Then  
READ PROTECT Account FROM ALL EXCEPT System
```

### **How to set up SSL**

**Aware IM** uses web server such as Tomcat or Weblogic to process web requests. The default web server that comes with **Aware IM** is Tomcat. If you want **Aware IM** to use SSL protocol for the web requests you can set up the web server to use SSL. Please refer to the documentation of the web server. See also the “Aware IM Installation Guide” document.

### **How to use existing database tables**

This is explained in detail in the “Working with Existing Database Tables” section in the Aware IM User Guide.

### **How to use data from several existing database tables**

Sometimes you may want to connect to existing database tables but display data from several existing database table in a single query. With normal Aware IM objects you can do it using shortcuts. With existing database tables you can also define a “normal” **Aware IM** object that would have references to different objects stored in the existing database tables and then define shortcuts to different attributes in these objects.

A better approach, however, is to define a view in the existing database that would select data from different tables, and then create a special **Aware IM** object that would point to this view.

### **How to integrate credit card payment system**

Many credit card payment systems, such as PayPal, require that your web site communicate with them via a particular URL. Whenever your web site needs to perform credit card payment the users need to click on the link with this URL, which will bring them to the web site of the credit card payment system. As part of the URL that your web site supplies to the credit card payment system you must also specify the URL that the credit card payment web site will use to return to your web site upon successful or un-successful completion of the payment.

If you want to integrate the credit card payment into your **Aware IM** system you need to configure the following:

1. Most credit card systems require that you pass certain parameters to the system as part of their URL, for example, PayPal system requires the following parameters (among others) – “amount”, “currency\_code”, “item\_name” etc. First of all you need to define a business object representing parameters of the payment with the attribute names being exactly equal to the names of the parameters you want to pass to the credit card payment system. For example, we could define PayPalPayment business object with “amount”, “currency\_code” and “item\_name” attributes.
2. Define an intelligent business object representing the credit card payment system, for example you can define the business object with the name PayPalSystem – see the “Defining Intelligent Business Objects” section in the Aware IM User Guide for details.
3. Define the URL channel as the default channel for this business object – see the “Setting Properties of URL Channel” section in the Aware IM User Guide. Specify the base URL of the credit card payment system as the “URL of the Service Provider” channel property. Specify the names of the parameters that the credit card payment system uses to identify the URL to return to as channel properties. For example, PayPal system requires that the parameter names for the successful and unsuccessful completion of payment are “return” and “cancel\_return” respectively. **Aware IM** will make sure that the values for these parameters are correctly set, so that the credit card system will return to the **Aware IM** system on successful or un-successful completion of the payment.
4. Define the payment service of this business object, for example “ProcessPayment” – see the “Defining Services of Intelligent Business Object” section in the Aware IM User Guide for details. Specify the business object you created in step 1 as the service input. Select “Standard Reply” as the service reply.
5. Define a process that will initialize the instance of the business object representing parameters of the payment (for example, PayPalPayment) request the service using the REQUEST SERVICE action, for example:

```
CREATE PayPalPayment WITH PayPalPayment.amount=100, ...  
REQUEST SERVICE ProcessPayment of PayPalSystem USING  
PayPalPayment
```

6. Define a menu item that will start the above process – see the “Setting Menu Item Properties” section in the Aware IM User Guide. Alternatively you can set up a form operation or a hyperlink on some presentation to start the process – see the “Adding/Editing Form Operations” and “Hyperlinks” section in the Aware IM User Guide. When the process starts **Aware IM** will navigate to the web site of the payment system. After the payment has been completed the web site of the credit card payment system will return to the **Aware IM** system (**Aware IM** will automatically display whether the payment request has been fulfilled).

### **How to communicate with another software system**

This is explained in the “Communication with Other Systems” section in the Aware IM User Guide.

### **How to communicate with a hardware device**

This is explained in the “Communication with Other Systems” section in the Aware IM User Guide.

### **How to expose a web-service**

Aware IM will automatically expose the services you configure for your system as web services, so that other software systems or web sites can use them – see the “Adding/Editing Services” section in the Aware IM User Guide.

### **How to expose a REST-ful service**

Aware IM can expose a service you configure either as a SOAP-based web service or as a REST-ful service or both. For REST-ful services exposed by Aware IM applications that need to call the service need to call a particular URL using the HTTP protocol. This URL should include the Aware IM server name and port where Tomcat listens to requests and also the name of the service to be called and parameters. The default URL where Aware IM will listen to REST-ful requests has the following format:

<http://ServerName:ServerPort/AwareIM/REST/BusinessSpace/ServiceName?parameters>

Parameters are specified as  
paramName1=paramValue1&paramName2=paramValue2&...

Where paramName1, paramName2 etc are names of the attributes of a business object that is specified as input to the process implementing the service. You can also specify your own custom URL in which case you also need to supply a file called rest.props

All the details are explained in the “Adding/Editing Services” section in the Aware IM User Guide

### **How to consume a REST-ful service**

To do this you need to define an intelligent business object by defining the REST communication channel with the object. This is described in the “Defining Intelligent Business Objects” section in the User Guide.

### **How to create a mobile version of a desktop application**

Starting from version 7.0 all features available for desktop applications are available for mobile applications as well and the other way around. However, because real estate on mobile platforms is severely limited you may want to create a special version for mobile phones and/or tablets that will use slightly different features compared to the desktop application.

To create a mobile version of your application, first create a special visual perspective for a particular access level. Select the “platform support” for this perspective to be “mobile phones” or “phones and tablets”. This perspective should then start all other queries, forms and processes of the mobile application. Most of the time you will just need to create a variation of your regular form, query, process or a visual perspective to be used in a mobile application. For more details see the Aware IM For Mobile Devices document.

### **How to display thumbnails – images with descriptions that the user can tap on**

This feature is very often used in mobile applications. To create a thumbnail view use a custom presentation of query results. See the “Displaying Query Results” section in the User Guide (“Custom Presentation” – thumbnail view presentation template). You can use either the “custom template” option or a “custom mobile template” option. The latter uses a special widget that supports grouping, infinite scrolling checkbox selection and operations.

If you use the “custom template” you will be able to use a “scroll view”, which is very popular for mobile applications – it displays one image at a time and users can swipe to the next or previous image.

### **How to display grids – lists with column names**

Just use regular grids (Queries with standard presentation). If your query uses inline editing or filtering make sure you tick the “Mobile rendering” option for the query (available through the More button)

### **How to define mobile application menus**

Usually for a mobile application running on a phone you would define two menus – the top toolbar and the slide-out menu (of course, both are optional). Defining the toolbar is simple – you just define a menu of the Toolbar type in the Top Bar frame of your visual perspective. Most likely you will place at least two buttons on this toolbar – the button that slides out the main menu and the button that implements the “Go back” functionality.

To define the “Back button” just define an operation of the “Go Back” type and add it to the toolbar. To define a button that slides out the main menu define this menu for the left or right frames first (the menu must be of the Plain List type). Then define the operation of the “Slide-In Left Frame” or “Slide-In Right Frame” types and add this operation to your toolbar.



You will probably have some real estate left on the toolbar. You can use this real estate for context operations – for example, to display Previous/Next buttons to traverse a form or some other operations invoked from a form or a query. In order to do this you need to designate the “Top Application Menu” as location for your form or query operations (see the Adding/Editing Panel Operations section in the User Guide for more details). You can also use a dynamic toolbar at the bottom for such operations. The toolbar will only appear when the corresponding form or query is displayed.

### **How to display forms**

You can just use regular forms for business objects. Some of the features of business object forms that you may find useful on mobile devices are listed below:

- Specify label with as percentage rather than absolute value
- Display Yes/No attributes as “on/off switches”
- Use native browser controls for dates and times – some mobile browsers have very rich and good looking mobile-specific in-built controls for entering dates and timestamps. You can use them in your forms instead of the more desktop oriented **Aware IM** in-built controls.
- Mobile style tabs – if the form has several form sections the tabs are displayed. You can select the “mobile style tabs” option to display tabs using mobile-friendly style – with larger font and top icon alignment. The option is available in the Navigation dialog of the Form Editor
- Mobile style of the form caption – most of the time you won’t display the default panel header for your mobile forms, because it uses too much real estate. You will either display no header at all or you will display a caption of the form in the middle of the screen. Selecting the “mobile style header” option in the Panel Header dialog will achieve that

### **How to display “gallery view”**

A popular feature on mobile devices is to show one image at a time with the user being able to swipe back and forth to display the next or previous image. To implement this feature use Queries in Custom Form (see Adding/Editing Queries in the User Guide), select the “Use Custom Data Template” option and then tick the “Use mobile scroll view” checkbox

### **How to upload an image from camera or photo library of the phone**

One approach is to define an attribute of the Picture type for one of your forms. **Aware IM** will automatically generate controls to upload the image from a camera or photo library. This approach will work in both browser-based mobile applications and native mobile applications.

Another approach is to use the `MOBILE CAMERA SNAP INTO` action in some process (or `MOBILE CAMERA GET INTO` action to get access to photo library). When Aware IM executes this action the user will be able to upload a photo from

camera or a photo library. The image will be written into the picture attribute specified in the action. For example:

```
MOBILE CAMERA SNAP INTO MyObject.MyPictureAttribute
```

Note that these actions will only work for native mobile applications – they will not work inside the mobile browser. Please see the User Guide and Rule Language Guide for more details about these actions.

### **How to display a picture full screen**

The quickest way is to define an attribute of the Picture type in some object and then define a special form with one form section and one cell showing the Picture attribute. For more details see the Aware IM For Mobile Devices document.

### **How to display a PDF document**

Just define an attribute of the Document type to store your PDF document. **Aware IM** will automatically generate controls to view the document on mobile devices at runtime.

### **How to display a Google Map**

You can use either static HTML for some content panel of a visual perspective or an object form – this is no different from displaying Google Maps for desktop applications.

### **How to determine and watch location of the current user**

It may be necessary for your application to determine the location of the current user – for example, if it needs to calculate users or places of interest closest to the current user.

The location of the current user can be determined by using the `MOBILE GET LOCATION INTO` action of the Rule Language. Before you can use this action, though, you need to add a special `MGeoLocation` object into your business space version. To do this right click on the Business Objects in the Configuration Elements Tree and select the “Add GeoLocation” object from the popup menu. Then you can use the `MOBILE GET LOCATION INTO` action that will write the current longitude and latitude (and some other values) of the current user into the attributes of the `MGeoLocation` object. For example:

```
CREATE MGeoLocation
MOBILE GET LOCATION INTO MGeoLocation
DISPLAY MESSAGE 'Longitude is ' +
MobileGeoLocation.Longitude
```

You can use this instance of the `MobileGeoLocation` object to display the location on a GoogleMap, for example. Or you can provide rules that calculate the distance between geo locations using the `DISTANCE` function.

You can also record changes to the user location. This can be done by using the `MOBILE START LOCATION WATCH INTO` action. This action will automatically track changes to the user location and write them into the provided instance of the `MobileGeoLocation` object. For example:

```
MOBILE START LOCATION WATCH INTO MGeoLocation
```

Tracking changes can be stopped using the `MOBILE STOP LOCATION WATCH FROM` action. For example:

```
MOBILE STOP LOCATION WATCH FROM MGeoLocation
```

Note that all geo location actions can be used in both browser-based mobile and non-mobile applications as well as in native mobile applications.

### **How to build native mobile applications**

This is explained in the “Building Native Mobile Applications” section in the Mobile Applications document. See also a video tutorial at <https://youtu.be/iOymwDel7oQ>

### **How to remember user credentials to prevent multiple logins**

On mobile phones a user may want to login only the first time he/she accesses an application and then the application should remember the user credentials, so that the next time the user accesses the application he/she does not have to login again. Aware IM provide a form called logonM.html that includes a checkbox called “Remember me”. If the user ticks this checkbox he/she will not have to login again. The default URL for this form is:

[http://localhost:8080/AwareIM/logonOp.aw?domain=BusinessSpaceName&mobile\\_login=true](http://localhost:8080/AwareIM/logonOp.aw?domain=BusinessSpaceName&mobile_login=true)

To force the browser to display a login form after the user has already been remembered add the “auto\_login=false” to the URL:

[http://localhost:8080/AwareIM/logonOp.aw?domain=BusinessSpaceName&mobile\\_login=true&auto\\_login=false](http://localhost:8080/AwareIM/logonOp.aw?domain=BusinessSpaceName&mobile_login=true&auto_login=false)

## **How to send push notifications to a mobile phone**

Sending push notifications to mobile phones is only supported for native mobile applications. If you want to send push notifications to mobile phones in your native mobile applications there are several major steps involved. Most of them need only to be done once, so don't be put off by the details described below.

1. Setup Apple and Google pre-requisites (for iOS and Android phones) – this should only be done once.
2. Enter these pre-requisites into Aware IM (also done once only)
3. Configure push notification rules in Aware IM

We will cover each of these steps in detail.

### ***Step 1 (Apple). Setup iOS pre-requisites***

- In order to use iOS push notifications you need a certificate.
- In order to build native iOS applications using PhoneGap you also need a certificate and a provisioning profile file.
  - a) Before you can generate certificates and provisioning profile you need to enter your Apple Developer account here <https://developer.apple.com/account#/welcome>
  - b) You can just sign in using your Apple ID.
  - c) Then you need to join Apple Developers Program here:
  - d) <https://developer.apple.com/programs/>
  - e) Once you have enrolled into this program you can generate certificates and provisioning profile for the PhoneGap build. Follow the instructions here: <https://coderwall.com/p/eceasa/getting-everything-for-building-ios-apps-with-build-phonegap-com>
  - f) To generate the certificate. Follow the instructions here: [https://developer.apple.com/library/ios/documentation/IDEs/Conceptual/AppDistributionGuide/AddingCapabilities/AddingCapabilities.html#//apple\\_ref/doc/uid/TP40012582-CH26-SW11](https://developer.apple.com/library/ios/documentation/IDEs/Conceptual/AppDistributionGuide/AddingCapabilities/AddingCapabilities.html#//apple_ref/doc/uid/TP40012582-CH26-SW11)

### ***Step 1 (Google). Setup Android pre-requisites***

- Create a project here <https://console.firebase.google.com/>
- Create an Android application for your project
- Go to the Application Settings (General tab) and download the file google-services.json. Keep this file in a safe location
- Go to the Cloud Messaging tab in application settings and copy the Services Key

### ***Step 2. Enter iOS and Android pre-requisites in Aware IM***

- In the Configuration Tool, double click on the Business Space Version you want to change and click on "Push Notifications" property in the list of version properties.
- A dialog will popup. In the "APNS" section of this dialog enter the full path to the iOS certificate file on your machine and certificate password.
- In the FCM Settings section of the dialog enter the location of the `google-services.json` file and enter the Services Key
- Once you click OK Aware IM will automatically add one predefined business object (`MobilePhone`) and one predefined notification `MobilePushNotification` to your business space version.

### **Step 3. Configure push notification rules in your application**

a) First you need to make sure that the owner of the phone subscribes to push notifications. You will need a process for this. The process should use the `MOBILE SUBSCRIBE` action. After this action has been executed a new instance of the `MobilePhone` object will be added to the context of the process. It is this object that the notification will be sent to. So most of the time it makes sense to associate this object with the logged in user. You can add a reference to the user object that will store one or several phones that the user owns. The process can then look like this:

```
MOBILE SUBSCRIBE
LoggedInRegularUser.MobilePhone = MobilePhone
(or INSERT MobilePhone in LoggedInRegularUser.Phones)
```

b) To send push notifications you need to create another process that uses the `MOBILE PUSH` action. The process should find all `MobilePhone` objects that it wants to send the notifications to and use the `MobilePushNotification`. For example:

```
FIND ALL MobilePhone
CREATE MobilePushNotification WITH
MobilePushNotification.Title='Test',MobilePushNotification,Subject='This is my test notification',MobilePushNotification.Sound='default'
MOBILE PUSH MobilePushNotification TO MobilePhone
```

### **How to integrate custom Cordova components in native mobile applications**

Cordova plugins are components that provide access to some built-in features of mobile phones, (such as camera or contacts), for which there is no Javascript access. When components are integrated into the system these features become available through some special Javascript functions that the plugin makes available to the developer. Cordova plugins can only be used in native mobile applications.

Aware IM integrates some Cordova plugins out-of-the-box and provides rule actions that activate them (for example, MOBILE PUSH or MOBILE CAMERA SNAP INTO). However, there are many plugins around and it is impossible to integrate all of them into Aware IM.

Still there is a way to do this by adding some custom Javascript to your application. This is described in detail in the Programmers Reference Guide

### **How to use a web-service**

If you want to use an existing web service exposed by some web site then you need to do the following:

1. Define an intelligent business object representing the web site exposing the web service with System intelligence type – see the “Defining Intelligent Business Objects” section in the Aware IM User Guide.
2. Define the SOAP channel as the default channel for this object. Specify the URL where the WSDL file describing the web service is exposed as the property of the channel – see the “Setting Properties of SOAP Channel” section in the Aware IM User Guide (alternatively ask the web service provider for the WSDL file, copy this file to a local file system and specify the URL to this, file, for example: `file:///c:/temp/lookup.wsdl`).
3. Discover services of the business object – see the “Discovering Services” section in the Aware IM User Guide. The web service you want to use should appear in the list of services exposed by the object.
4. Define a rule that will request the service of the business object using the `REQUEST SERVICE` action.

### **How to register Aware IM application in Facebook**

In order to allow users to login via Facebook you need to register your Aware IM application with Facebook. In order to do this:

1. Go to the URL <https://developers.facebook.com/apps>
2. Create a new application
3. When specifying site URL for the application enter the following URL:  
<http://YourServerName:8080/AwareIM>

You can use `localhost` instead of the server name when testing your application.

When specifying “redirect URI” of the application enter the following URI:  
<http://YourServerName:8080/AwareIM/req.awfb2>

4. Copy application id and secret that Facebook should assign to your application – you will need these when setting up login options (see [How to setup login options via Facebook, Twitter and Google](#) section)

When users login to your Aware IM application they should use the following URL:

<http://YourServerName:8080/AwareIM/logon.awfb?domain=YourBusinessSpaceName&testingMode=false>

### **How to register Aware IM application in Twitter**

In order to allow users to login via Twitter you need to register your Aware IM application with Twitter. In order to do this:

1. Go to the URL <https://dev.twitter.com/apps>
2. Create a new application
3. When specifying callback URL for the application enter the following URL:  
<http://YourServerName:8080/AwareIM>  
You can use `localhost` instead of the server name when testing your application.  
When specifying “redirect URI” of the application enter the following URI:  
<http://YourServerName:8080/AwareIM/req.awtw2>
4. Copy consumer key and secret that Twitter should assign to your application – you will need these when setting up login options (see [How to setup login options via Facebook, Twitter and Google](#) section)

When users login to your Aware IM application they should use the following URL:

<http://YourServerName:8080/AwareIM/logon.awtw?domain=YourBusinessSpaceName&testingMode=false>

### **How to register Aware IM application in Google**

In order to allow users to login via Google you need to register your Aware IM application with Google. In order to do this:

1. Go to the URL <https://code.google.com/apis/console>
2. Create a new application
3. When specifying redirect URI for the application enter the following URI:  
<http://YourServerName:8080/AwareIM/req.awggl2>  
You can use `localhost` instead of the server name when testing your application.  
When specifying “redirect URI” of the application enter the following URI:  
<http://YourServerName:8080/AwareIM/req.awggl2>
4. Copy client id and secret that Google should assign to your application – you will need these when setting up login options (see [How to setup login options via Facebook, Twitter and Google](#) section)



5. If using synchronization with the Google Calendar make sure that access to the calendar API is allowed!

When users login to your Aware IM application they should use the following URL:

<http://YourServerName:8080/AwareIM/logon.awggl?domain=YourBusinessSpaceName&testingMode=false>

### **How to re-use configurations**

On certain occasions you may want to integrate all or parts of configurations of other systems into your system to re-use the definitions of business objects, business rules and other elements. For example, you might want to integrate certain aspects of sample applications into your system. To do this:

1. Create a business space representing the system you want to re-use – see the “Adding/Editing Business Spaces” section in the Aware IM User Guide.
2. Import the configuration you want to re-use from a BSV file – see the “Importing Business Space Version” section in the Aware IM User Guide.
3. Copy and paste the elements you want to re-use into your system (see the “Copying and Pasting Elements” section in the Aware IM User Guide)
4. Make the necessary adjustments to the copied elements if required.
5. Check integrity of the version to make sure that it remains consistent – see the “Checking Version Integrity” section in the Aware IM User Guide.

### **How to import/export data**

This is explained in detail in the “Export and Import” section in the Aware IM User Guide.

### **How to define import/export templates to map data**

You can get end users to define their own import and export templates that describe data mapping between their data files and Aware IM attributes. For this you need to add the command of the “Manage Import/Export Templates” type to your menu. For more details about import and export templates please watch this [video tutorial](#).

### **How to set up periodic processes**

If you want the system to perform certain actions periodically without any user interaction (for example, regularly delete temporary records) then you need to configure scheduling rules – see the “Scheduling” section in the Aware IM User Guide for details.

If you want periodic actions that involve interaction with the user (for example, to periodically inform the user about important events in the system) then you need to setup a “background process”. To do this define a process to perform the work, go to the General tab of the corresponding visual perspective and setup this process as a background process.

### **How to extend Aware IM**

If you are a programmer then you can easily write programming components that can extend the functionality of your **Aware IM** system – for example, you can plug-in new functions, new document types, new communication channels etc. See the “Aware IM Programmer’s Reference” document for details.

### **How to get users to define their own workflows**

Sometimes developers who configure applications cannot predict how their customers will use the system – some of them may have unique requirements, which change dynamically. By letting end users define their own workflows developers can make the system extra flexible. In order to get users to define their own workflows the configurator has to add the command of the “Manage User Defined Processes” to the application menu. There are also other types of commands that can be useful for this module, such as Configure User Defined Processes, Manage Events and a few others. For more details about this powerful module please watch this [video tutorial](#).

### **How to support offline mode**

“Offline mode” is when users work with an application without Internet connection. This is explained in detail in this [video tutorial](#).

### **How to count web site visitors**

Whenever a new user logs into the system **Aware IM** sends the special login notification. You can attach rules to this notification to count the number of users (see the “Handling Login Events” section in the Aware IM User Guide). For example, you can define a business object representing the statistics of your web site and define an attribute in this object that will count the number of visitors. The rules attached to the login notification will increment the value of this attribute.

### **How to track activity of the registered users**

Whenever a new user logs into the system **Aware IM** sends the special login notification; whenever a user logs out **Aware IM** sends the logout notification. You can attach rules to these notifications to track the activity on your web site (see the “Handling Login Events” and “Handling Logout Events” sections in the Aware IM User Guide). For example, you can define a business object

representing the statistics of your web site and define attributes in this object that will count the number of current users, the number of visitors etc. The rules attached to the login and logout notifications will set the values of these attributes accordingly.

### **How to create applications in different languages**

This is explained in the “Creating applications in different languages” section of the User Guide. Here is a summary:

- a) If you want to run Aware IM applications in a single non-English language (or a variation of the English language) you need to do the following:
  - Define your configuration strings (attribute labels, messages used in DISPLAY MESSAGE or REPORT ERROR rules etc) using the language of your choice
  - Define a locale. When creating a locale Aware IM will automatically extract all your configuration strings and if you defined them in the language of your choice you don't need to translate them
  - Import system strings from a file (if a file exists for your language)
  - Translate the remaining system strings by clicking on the “Translation” column in the locale editor
  - Mark the locale as “default”
- b) If you want to run **Aware IM** applications in multiple languages you need to do the following:
  - Define a locale for every language you want to use. Provide the translation for your configuration and system strings for every language
  - Define a rule attached to your user object (either RegularUser or some other member of the SystemUser group) to assign an appropriate locale to the user, for example:

```
IF RegularUser.Country = 'Germany' Then  
    RegularUser.Locale = 'German'
```

### **How to optimise application for best performance**

Out-of-the-box **Aware IM** offers a pretty good performance for many applications. However, it does not mean that you should always rely on Aware IM to deliver the best performance. A lot depends on how you write your application as well. The following areas are important when you write an application:

- Performance of business rules
- Performance of queries
- How quickly each request from the user is processed in your system

Below are some recommendations on how to improve performance in each of these areas:

- 1) Check the Log Viewer for the execution of rules that refer to the object (cross reference rules). These may be expensive. Execution of many of them may be unnecessary. Turn them off by going to the Advanced tab of the rule and ticking "Do not execute referred" (or something like that - the last checkbox there)
- 2) Aggregate operations (SUM, COUNT etc) can be expensive if you have lots of data. Try not to use them if you can
- 3) Unconditional rules can be expensive (especially if they use aggregate operations). Make them conditional (using WAS CHANGED expression, for example)
- 4) Check for repetitive execution of rules. The rule engine can execute rules repetitively if the next rule triggers a condition that causes the previous rule to fire again. Optimise this carefully. Use rule priorities to make sure that expensive rules are only executed once
- 5) If you find that a particular query does not perform well replace it with a stored procedure (Aware IM allows you to use stored procedures – see the User Guide for more details)
- 6) If you need to support many concurrent users make sure that there are no CPU-extensive operations in your application. Or make sure that the CPU-extensive operations are executed at night or by a separate server.
- 7) Add additional servers to scale up your system if necessary (Aware IM does support scaling and load balancing)

### **How to search for multiple objects**

Sometimes you may need to search for instances of different business object based on the values of attributes common to all of them. For example, let us assume you have defined SavingAccount and CreditAccount objects. Both objects contain different attributes and yet both of them have the Balance attribute defined. Suppose you want to find all accounts (saving and credit) with the balance greater than 1000 dollars.

The solution is to add both objects to a business group (called Account, for example) – see the “Adding/Editing Business Object Groups” section in the Aware IM User Guide – and define a query that will look for the business object group rather than an individual business object (see the “Adding/Editing Queries” section in the Aware IM User Guide).

### **What to do if an object can be related to multiple objects**

Sometimes when you define relationships between business objects you may want to establish a relationship with different business object through the same reference attribute. For example, let us assume that you are defining an Account object and you want to establish a relationship between the account and its transactions by defining the MyTransactions reference attribute. However, you have several business objects that represent transactions – WithdrawalTransaction and DepositTransaction. These transactions have some different attributes and some common attributes (for example Amount).

The solution is to add different objects representing transactions into a business object group (called Transaction, for example) - see the “Adding/Editing Business Object Groups” section in the Aware IM User Guide – and get the reference attribute to refer to a business object group rather than the individual business object (see the “Setting Properties of Reference Attributes” section in the Aware IM User Guide).

### **How to display currency symbol**

If some attribute of a business object represents currency, then you may want to display the value of the attribute with the currency symbol in front of the value – for example, \$400. To do this define the attribute to be of the Number type and add the currency symbol to the format of the number – for example, “\$#” - see “Attributes of Number Type”, Appendix C in the Aware IM User Guide. The value of the attribute will be displayed with the currency symbol on all forms. If you want to display the currency symbol only in certain reports or presentations, you should not change the format of the attribute, but instead include the currency symbol inside the tags of the reports or presentations, for example:

```
<<Account.Balance, $#.00>>
```

### **How to generate unique ID for the attribute**

In many cases it is necessary to have a number uniquely identifying business elements, such as customer number, order number, account number, etc. If you have no particular requirements on the numbering of your objects you can simply use the ID attribute.

If you need to control the numbering of your business objects you will need to configure the attribute for this purpose of the Number type and mark it as “auto-incremented”

### **How to compare strings ignoring case**

Use `TO_LOWER_CASE` or `TO_UPPER_CASE` functions to convert both strings to upper or lower case and then compare them. See also the “Text Functions” section in the Aware IM User Guide.

### **How to force users to enter parameters of the search**

If you want a user to enter parameters of a search you need to specify a query that requires some of its parameters to be entered at run-time by the user (see the “Configuring Queries” section in the Aware IM User Guide). Sometimes you may want to force the user to enter a parameter. If she does not you do not want the query to run. The query parameters mechanism does not support mandatory parameters (if parameters are not entered the corresponding query condition is ignored). To force entering of query parameters you can define a temporary business object (with Persistence flag set to “Not persisted” – see the “Adding/Editing Business Objects” section in the Aware IM User Guide) with attributes that correspond to the parameters of the search. You can make some of the attributes mandatory by marking them as “value must be provided” – see the “Common Properties” section in the Aware IM User Guide). Then you need to define a process that will let the user enter the temporary object and then run a query. For example:

```
ENTER NEW QueryParametersObject
FIND Account WHERE
Account.Status=QueryParametersObject.Status
```