

openCRX Quick Start

Version 1.6.0

www.opencrx.org

openCRX Quick Start: Version 1.6.0

by www.opencrx.org

The contents of this file are subject to a BSD license (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License *here* (<http://www.opencrx.org/license.htm>)

Table of Contents

| | |
|---|-----------|
| 1. About this Book | 1 |
| Who this book is for | 1 |
| What do you need to understand this book | 1 |
| 2. Prerequisites | 2 |
| Enabling Outgoing E-mail (SMTP Adapter) | 2 |
| 3. Create Logins..... | 4 |
| Overview | 4 |
| Initial Setup..... | 5 |
| Creation of a new data Segment..... | 9 |
| Create a User..... | 13 |
| 4. Managing the Provider List..... | 19 |
| 5. Security | 21 |
| Security Setting of New Objects | 21 |
| Resetting Security | 21 |
| 6. Importing Data | 22 |
| Importing vCard Files..... | 22 |
| Importing iCalendar Files | 22 |
| Importing XML Files | 23 |
| 7. Exporting Data | 25 |
| Exporting vCard Files..... | 25 |
| Exporting iCalendar Files | 25 |
| Exporting XML Files | 26 |
| 8. The Event Subscription / Notification Service | 28 |
| Example Subscription – New Accounts | 29 |
| 9. Outbound E-mail Service | 32 |
| Default Mail Server | 34 |
| 10. Integration of MS Word | 35 |
| 11. Next Steps | 37 |
| A. Appendix..... | 38 |
| Bibliography | 39 |

List of Figures

| | |
|--|----|
| 2-1. Secure communication between openCRX server and multiple mail servers | 3 |
| 3-1. Data in the openCRX database can be partitioned into data segments..... | 5 |
| 3-2. Initial login as admin-Root | 6 |
| 3-3. Initial screen after login as admin-Root..... | 8 |
| 3-4. Realm "Default" is created automatically | 8 |
| 3-5. admin-Root sets Access Levels of Codes | 9 |
| 3-6. admin-Root sets Access Levels of Uoms..... | 9 |
| 3-7. Create a new Segment | 10 |
| 3-8. Verify creation of new principals admin-Standard and loader-Standard | 10 |
| 3-9. Login as admin-Standard..... | 11 |
| 3-10. Homepage of admin-standard (charts initialized)..... | 12 |
| 3-11. Verify Users and User Groups | 12 |
| 3-12. Create a new User by clicking on the button [Principal]..... | 13 |
| 3-13. Enter Principal name (=Login name) into the field "Qualifier"..... | 13 |
| 3-14. Add Principal to Principal Group - Step 1 | 14 |
| 3-15. Add Principal to Principal Group - Step 2 | 14 |
| 3-16. Add Principal to Principal Group - Step 3 | 15 |
| 3-17. Add Principal to Principal Group - Step 4 | 15 |
| 3-18. As admin-Standard, create a Contact for the new user "guest" | 16 |
| 3-19. Operation [Create User]..... | 16 |
| 3-20. As root, verify creation of new user "guest" | 17 |
| 3-21. The new user "guest" can login in the standard GUI..... | 18 |
| 4-1. Provider List | 19 |
| 6-1. Operation vCard import..... | 22 |
| 6-2. Result of the vCard import operation | 22 |
| 6-3. Operation iCalendar import | 23 |
| 6-4. Result of the iCalendar import operation..... | 23 |
| 6-5. Operation Import on UserHome..... | 24 |
| 7-1. Operation vCard export | 25 |
| 7-2. Result of the vCard export operation..... | 25 |
| 7-3. Operation iCalendar export..... | 26 |
| 7-4. Result of the iCalendar export operation | 26 |
| 7-5. Operation Export XML..... | 27 |
| 8-1. Overview Event Subscription / Notification | 28 |
| 8-2. Topics included in the openCRX distribution | 29 |
| 8-3. Create a new subscription | 30 |
| 8-4. Lookup Inspector for Topics..... | 30 |
| 8-5. Subscription "New Account" | 31 |
| 8-6. UserHome with the newly created subscription "New Account" | 31 |
| 9-1. Create a new E-mail Account..... | 32 |
| 9-2. E-mail Account Object | 32 |
| 9-3. Grid Tab [E-mail]..... | 33 |
| 9-4. E-mail Options | 33 |
| 9-5. Create a Default Mail Server E-mail Account..... | 34 |
| 10-1. Ready to launch MS Word..... | 36 |

List of Examples

| | |
|---|----|
| 3-1. openCRX.users.properties with user=password syntax..... | 4 |
| 3-2. openCRX.roles.properties with user.Roles=role1,role2 syntax..... | 4 |
| 3-3. initializing the openCRX Root servlet..... | 6 |
| 4-1. web.xml with the provider declaration for Forecasts..... | 20 |
| 4-2. web.xml with the Forecasts provider commented out..... | 20 |
| 10-1. Sample VBA code (sample-letter.doc)..... | 35 |
| 10-2. additionalElementDefinition for MS Word Sample Letter..... | 35 |

Chapter 1. About this Book

This book describes what you need to get started with *openCRX* and how you configure *openCRX*.

Who this book is for

The intended audience are *openCRX* administrators and advanced users.

What do you need to understand this book

This book describes how to install *openCRX* and how to configure *openCRX*. If you intend to install *openCRX* it is a plus if you are comfortable with application servers and database servers (even though the explanations in this guide are rather detailed).

Chapter 2. Prerequisites

In a first step you must decide which database and which application server you are going to use.

As far as the **database** is concerned, the *openCRX FAQ* (<http://www.opencrx.org/faq.htm#db>) might give you some guidance in making your choice (please note that the Open Source distribution of *openCRX* includes all the required configuration/deployment files for *MySQL* (<http://www.mysql.com/>), *MaxDB* (<http://www.mysql.com/products/maxdb/>), *PostgreSQL* (<http://www.postgresql.org/>), *MS SQL* (<http://www.microsoft.com/sql/>), and *Oracle* (<http://www.oracle.com/>).

As far as the **application server** is concerned, your best bet is probably the one you know best as long as it is J2EE-compliant (additional information regarding the choice of an application server is available in the *openCRX FAQ* (<http://www.opencrx.org/faq.htm#appserver>)). The Open Source distribution of *openCRX* includes all the required configuration/deployment files for *JBoss* (<http://www.jboss.org/>) (which is also Open Source and free), *BEA Weblogic* (<http://www.bea.com/>), and *IBM WebSphere* (<http://www.ibm.com/websphere/>).

Having made your choices regarding database and application server you should get the appropriate *openCRX Installation Guides* from *here* (<http://www.opencrx.org/documents.htm>). Following those guides you should then install the *openCRX* database and the application server.

The remainder of this document assumes that you decided for MySQL and JBoss and hence you should have a working installation of

- the **MySQL** database (following the **openCRX Installation Guide for MySQL**) and
- the **JBoss** application server (following the **openCRX Installation Guide for JBoss**)

Enabling Outgoing E-mail (SMTP Adapter)

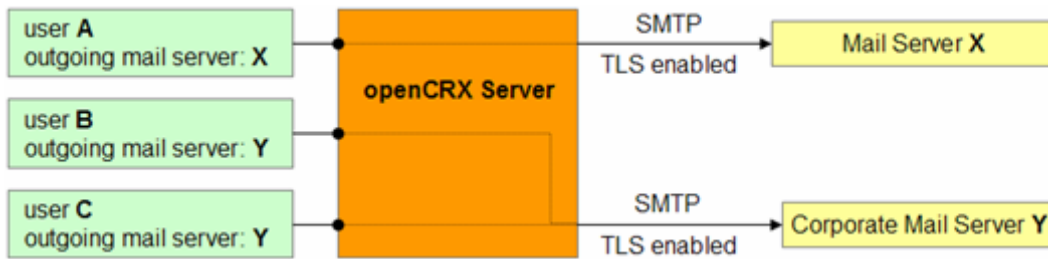
The following steps are optional and only required if you want to enable outgoing e-mail by activating the *openCRX* SMTP Adapter.

- download *J2EE JavaMail* (<http://java.sun.com/products/javamail/>) and copy the file *mail.jar* to the directory `JAVA_HOME/jre/lib/ext`
- download *JavaBeans Activation Framework (JAF)* (<http://java.sun.com/products/javabeans/glasgow/jaf.html>) and copy the file *activation.jar* to the directory `JAVA_HOME/jre/lib/ext`
- copy the file *opencrx-sendmail.jar* contained in the *opencrx-core* distribution (e.g. *opencrx-1.5.0-core.CRX.jre-1.4.zip*) to the directory `JAVA_HOME/jre/lib/ext`



Important Make sure that you add **JAVA_HOME** to your system environment variables, e.g.
`JAVA_HOME=D:\pgm\jdk1.4.2`

Figure 2-1. Secure communication between openCRX server and multiple mail servers



The openCRX SMTP Adapter works best in combination with a corporate mail server (depending on the volume of outgoing e-mail you might even consider a dedicated e-mail server) with **decent response times**. Every user can define his preferred mail server or rely on the *Default Mail Server* defined by the segment administrator. The *openCRX* SMTP Adapter is TLS enabled, i.e. if the mail server supports TLS then the communication between the *openCRX* server and the mail server will be encrypted. Configuration of e-mail options at the user level is explained in chapter *Outbound E-mail Service*.

Chapter 3. Create Logins

For the following steps we assume that the *openCRX* administrator has configured the users **admin-Root**, **admin-Standard** and **guest** with the appropriate roles on the application server. In the case of JBoss he would have edited the files *openCRX.users.properties* and *openCRX.roles.properties* in directory *.../jboss-4.0.1/server/default/conf* as follows and then restarted the application server:

Example 3-1. *openCRX.users.properties* with `user=password` syntax.

```
admin-Root=rootSecret
admin-Standard=adminSecret
guest=guest
```

Example 3-2. *openCRX.roles.properties* with `user.Roles=role1,role2` syntax.

```
admin-Root.Roles=OpenCrxRoot
admin-Standard.Roles=OpenCrxAdministrator
guest.Roles=OpenCrxUser
```



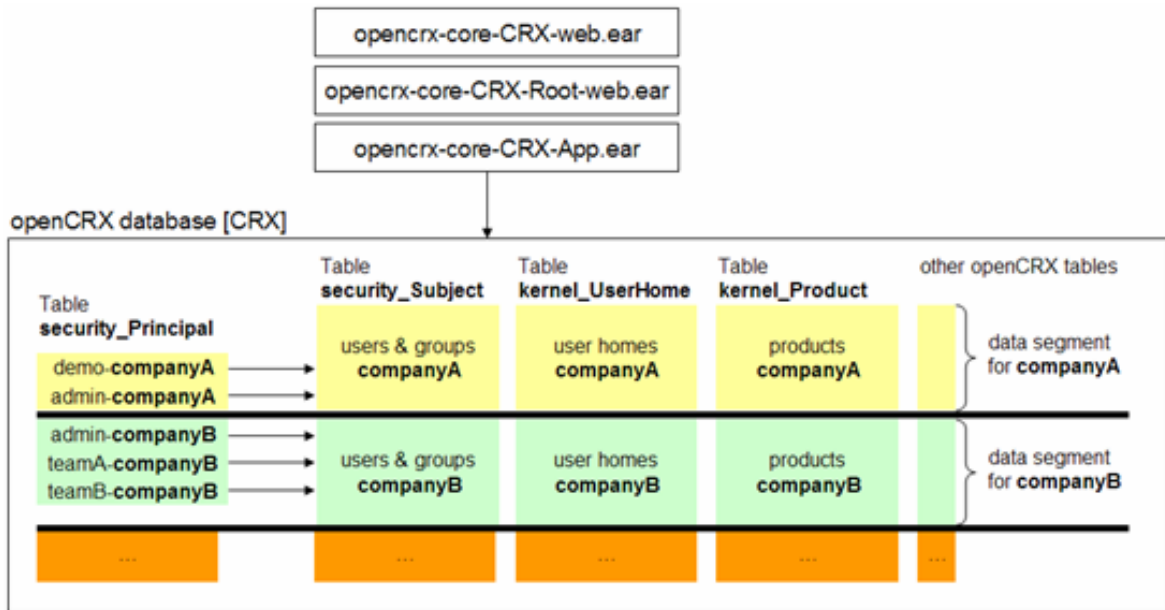
Important Before a user can login to *openCRX* you must first create a new login at the application server level. Of course you can automate this process. E.g. on JBoss you can replace the file-based *org.jboss.security.auth.spi.UsersRolesLoginModule* login module with the database login module *org.jboss.security.auth.spi.DatabaseServerLoginModule* and configure it to access the *openCRX* security tables *security_Principal* and *security_Credential*. Please refer to the JBoss installation guide if you want to make use of the database login module.

Overview

Before we get started with setting up *openCRX* it is helpful if you know that – by default – there are three types of users playing quite different roles in the context of *openCRX*:

- **Root** (default login is *admin-Root*)
 - **Performs the initial openCRX setup.** This initializes the database and loads basic *openCRX* working data (e.g. code tables). This is a one-time task.
 - **Creates new segments.** *openCRX* is multi-entity enabled. Each entity's data is stored in its own data segment. The segmentation of data is a basic and important concept of *openCRX*. It allows to setup private areas for different user groups, e.g. branches of a company or different small business companies. A user can have multiple logins, whereas a login allows access to exactly one segment. E.g. a user **demo** can have the logins **demo-companyA** and **demo-companyB** which are managed by root in the table **security_Principal**. If the user logs in as **demo-companyA** he/she has access to the segment **companyA**, i.e. he/she can only see data created by users of **companyA**. This is shown in *Figure 3-1*. A small to medium setup has typically 1 data segment only, named **Standard**.
- **Administrator** (default login is *admin-<SegmentName>*). Each data segment has its own administrator. E.g. the administrator for the segment **companyA** has the login **admin-companyA**, the administrator for the segment **Standard** has the login **admin-Standard**. The administrator is responsible for all administrative tasks related to a particular data segment, e.g. creating users.
- **User:** A user is a standard *openCRX* user which manages accounts, products, leads, activities, etc. Each user is assigned to a segment and is member of one or more user groups.

Figure 3-1. Data in the openCRX database can be partitioned into data segments.



Important All users who access *openCRX* by the same web application (e.g. *opencrx-core-CRX*) also share the same customization files (user interface, code tables, basic data). The openCRX/Core README explains how to setup multiple customized web applications.

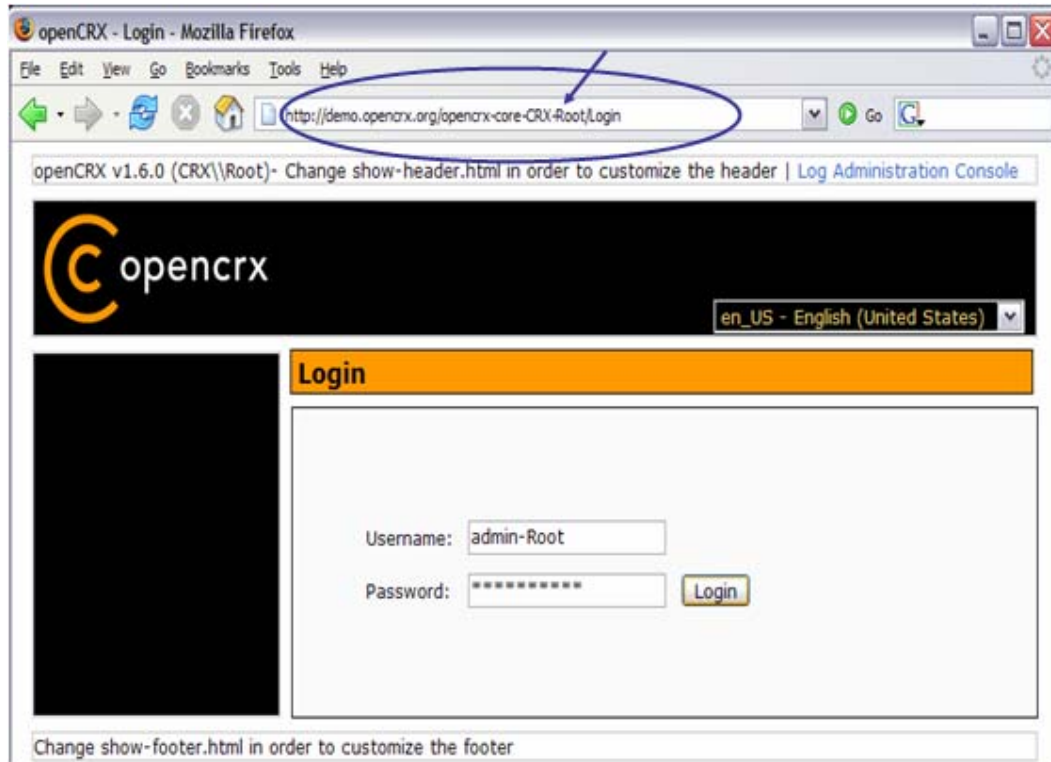
The following sections explain:

- *Initial Setup of openCRX*
- *Creation of a new data Segment*
- *Create a User*

Initial Setup

After installing *openCRX* on the application server and creating the logins you are now ready for the initial setup. Connect to the openCRX Root login page (e.g. <http://localhost:8080/openctx-core-CRX-Root/Login>). Login as **admin-Root** as shown in *Figure 3-2*.

Figure 3-2. Initial login as admin-Root



The servlet loads the initial data, e.g. default security policies, subjects and principals, units of measurement, code tables, etc. The output is shown on the application server console as shown below:

Example 3-3. initializing the openCRX Root servlet

```
09:16:57,207 INFO [STDOUT] Sun Feb 27 09:16:57 CET 2005: Login: requestURL=http://...
09:16:57,207 INFO [STDOUT] Sun Feb 27 09:16:57 CET 2005: Login: locale=null
09:17:11,818 INFO [STDOUT] inspecting /WEB-INF/config/ui/en_US
09:17:12,038 INFO [STDOUT] inspecting /WEB-INF/config/ui/de_CH
09:17:12,219 INFO [STDOUT] inspecting /WEB-INF/config/ui/es_MX
09:17:12,389 INFO [STDOUT] inspecting /WEB-INF/config/ui/zh_CN
09:17:12,609 INFO [STDOUT] inspecting /WEB-INF/config/ui/sv_SE
09:17:12,789 INFO [STDOUT] inspecting /WEB-INF/config/ui/tr_TR
09:17:12,970 INFO [STDOUT] inspecting /WEB-INF/config/ui/fa_IR
09:17:13,150 INFO [STDOUT] inspecting /WEB-INF/config/ui/fr_FR
09:17:13,330 INFO [STDOUT] inspecting /WEB-INF/config/ui/ru_RU
09:17:13,520 INFO [STDOUT] inspecting /WEB-INF/config/ui/pl_PL
09:17:13,771 INFO [STDOUT] loading /WEB-INF/config/ui/en_US/abstractcontract.xml
09:17:14,121 INFO [STDOUT] loading /WEB-INF/config/ui/en_US/account.xml
09:17:14,131 INFO [STDOUT] loading /WEB-INF/config/ui/en_US/activity.xml
09:17:14,131 INFO [STDOUT] loading /WEB-INF/config/ui/en_US/address.xml
09:17:14,161 INFO [STDOUT] loading /WEB-INF/config/ui/en_US/addressgroup.xml
09:17:14,161 INFO [STDOUT] loading /WEB-INF/config/ui/en_US/admin.xml
...
...
09:17:22,413 INFO [STDOUT] loading /WEB-INF/config/ui/pl_PL/userhome_wf.xml
09:17:22,423 INFO [STDOUT] loading /WEB-INF/config/ui/pl_PL/warehouse.xml
09:17:22,433 INFO [STDOUT] loading /WEB-INF/config/ui/pl_PL/wf.xml
09:17:27,591 INFO [STDOUT] storing 1718 ui elements
09:17:36,854 INFO [STDOUT] loading texts
```

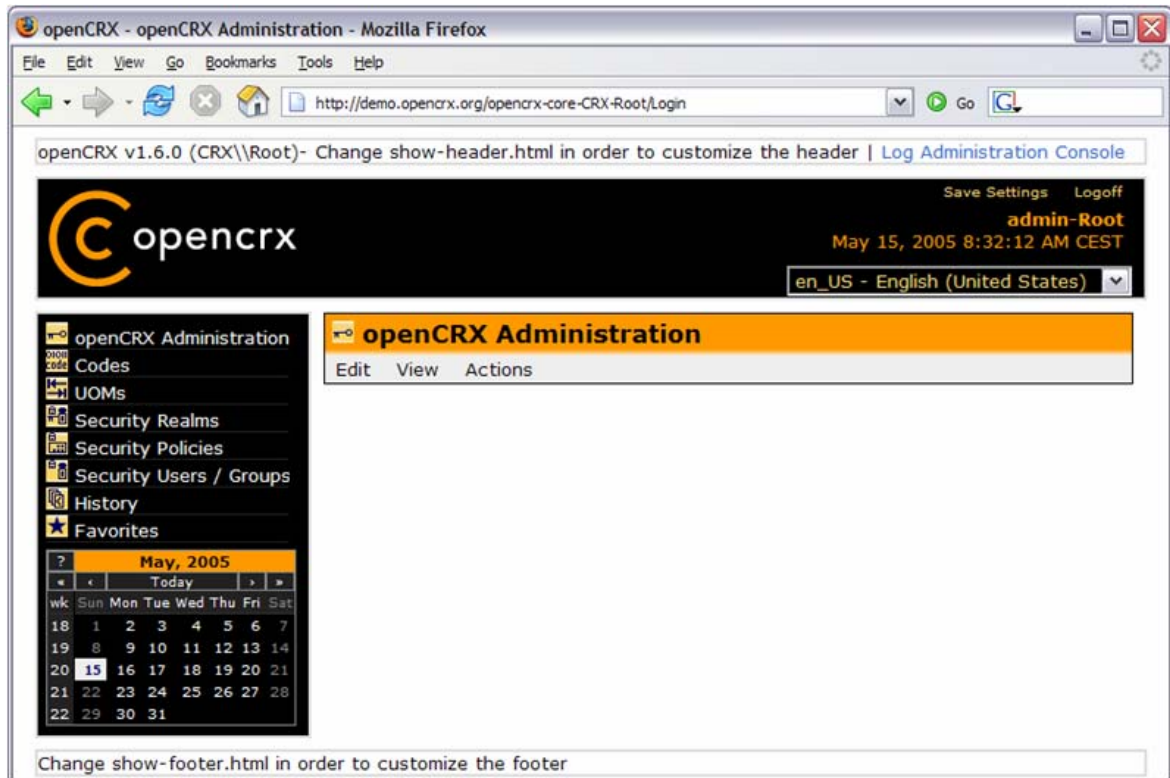
```

09:17:36,854 INFO [STDOUT] loading /WEB-INF/config/texts/en_US/texts.properties
09:17:36,854 INFO [STDOUT] loading /WEB-INF/config/texts/en_US/opencrx.texts.properties
09:17:36,864 INFO [STDOUT] loading /WEB-INF/config/texts/de_CH/opencrx.texts.properties
09:17:36,864 INFO [STDOUT] loading /WEB-INF/config/texts/de_CH/texts.properties
09:17:36,874 INFO [STDOUT] loading /WEB-INF/config/texts/es_MX/opencrx.texts.properties
09:17:36,874 INFO [STDOUT] loading /WEB-INF/config/texts/es_MX/texts.properties
09:17:36,874 INFO [STDOUT] loading /WEB-INF/config/texts/zh_CN/opencrx.texts.properties
09:17:36,884 INFO [STDOUT] loading /WEB-INF/config/texts/zh_CN/texts.properties
09:17:36,884 INFO [STDOUT] loading /WEB-INF/config/texts/sv_SE/opencrx.texts.properties
09:17:36,884 INFO [STDOUT] loading /WEB-INF/config/texts/sv_SE/texts.properties
09:17:36,894 INFO [STDOUT] loading /WEB-INF/config/texts/tr_TR/opencrx.texts.properties
09:17:36,894 INFO [STDOUT] loading /WEB-INF/config/texts/tr_TR/texts.properties
09:17:36,904 INFO [STDOUT] loading /WEB-INF/config/texts/fa_IR/opencrx.texts.properties
09:17:36,904 INFO [STDOUT] loading /WEB-INF/config/texts/fa_IR/texts.properties
09:17:36,904 INFO [STDOUT] loading /WEB-INF/config/texts/fr_FR/opencrx.texts.properties
09:17:36,914 INFO [STDOUT] loading /WEB-INF/config/texts/fr_FR/texts.properties
09:17:36,914 INFO [STDOUT] loading /WEB-INF/config/texts/ru_RU/texts.properties
09:17:36,924 INFO [STDOUT] loading /WEB-INF/config/texts/ru_RU/opencrx.texts.properties
09:17:36,924 INFO [STDOUT] loading /WEB-INF/config/texts/pl_PL/opencrx.texts.properties
09:17:36,924 INFO [STDOUT] loading /WEB-INF/config/texts/pl_PL/texts.properties
09:17:36,964 INFO [STDOUT] loading data
09:17:36,974 INFO [STDOUT] loading /WEB-INF/config/bootstrap/100_security_policy.xml
09:17:37,024 INFO [STDOUT] storing 5 objects
09:17:43,133 INFO [STDOUT] loading /WEB-INF/config/bootstrap/101_security_subjects.xml
09:17:45,346 INFO [STDOUT] storing 5 objects
09:17:45,476 INFO [STDOUT] loading /WEB-INF/config/bootstrap/102_security_realms.xml
09:17:45,526 INFO [STDOUT] storing 6 objects
09:17:45,737 INFO [STDOUT] loading /WEB-INF/config/bootstrap/200_code_segment.xml
09:17:46,989 INFO [STDOUT] storing 1 objects
09:17:48,881 INFO [STDOUT] replacing org::opencrx::kernel::code/provider/CRX/segment/Root
09:17:48,951 INFO [STDOUT] loading /WEB-INF/config/bootstrap/300_admin_segment.xml
09:17:51,545 INFO [STDOUT] storing 1 objects
09:17:51,565 INFO [STDOUT] done
09:17:51,575 INFO [STDOUT] loading codes
09:17:51,585 INFO [STDOUT] loading /WEB-INF/config/code/en_US/accesslevel.xml
09:17:51,595 INFO [STDOUT] loading /WEB-INF/config/code/en_US/accountcategory.xml
09:17:51,595 INFO [STDOUT] loading /WEB-INF/config/code/en_US/accountstate.xml
09:17:51,605 INFO [STDOUT] loading /WEB-INF/config/code/en_US/accounttype.xml
09:17:51,615 INFO [STDOUT] loading /WEB-INF/config/code/en_US/alertstate.xml
...
...
09:17:56,723 INFO [STDOUT] loading /WEB-INF/config/code/pl_PL/usageaddress_ext_phone.xml
09:17:56,743 INFO [STDOUT] loading /WEB-INF/config/code/pl_PL/usageaddress_ext_postal.xml
09:17:56,753 INFO [STDOUT] loading /WEB-INF/config/code/pl_PL/usageaddress_ext_web.xml
09:17:56,763 INFO [STDOUT] loading /WEB-INF/config/code/pl_PL/usageproductbaseprice.xml
09:17:56,763 INFO [STDOUT] loading /WEB-INF/config/code/pl_PL/utcoffset.xml
09:17:56,783 INFO [STDOUT] storing 1136 code entries
09:18:17,092 INFO [STDOUT] done
09:18:17,102 INFO [STDOUT] inspecting /WEB-INF/config/filters/
09:18:17,142 INFO [STDOUT] loading /WEB-INF/config/filters/
09:18:20,917 INFO [STDOUT] filter loaded for references 1943
09:18:20,917 INFO [STDOUT] loading data
09:18:20,917 INFO [STDOUT] loading /WEB-INF/config/data/uom_SI_and_Paper.xml
09:18:22,069 INFO [STDOUT] storing 39 objects
09:18:24,422 INFO [STDOUT] done
09:18:24,783 INFO [STDOUT] inspecting /WEB-INF/config/ui/en_US
09:18:24,963 INFO [STDOUT] inspecting /WEB-INF/config/ui/de_CH
09:18:25,113 INFO [STDOUT] inspecting /WEB-INF/config/ui/es_MX
09:18:25,254 INFO [STDOUT] inspecting /WEB-INF/config/ui/zh_CN
09:18:25,394 INFO [STDOUT] inspecting /WEB-INF/config/ui/sv_SE
09:18:25,544 INFO [STDOUT] inspecting /WEB-INF/config/ui/tr_TR
09:18:25,684 INFO [STDOUT] inspecting /WEB-INF/config/ui/fa_IR
09:18:25,824 INFO [STDOUT] inspecting /WEB-INF/config/ui/fr_FR
09:18:25,975 INFO [STDOUT] inspecting /WEB-INF/config/ui/ru_RU
09:18:26,125 INFO [STDOUT] inspecting /WEB-INF/config/ui/pl_PL
09:18:26,285 INFO [STDOUT] inspecting /WEB-INF/config/filters/

```

After the successful startup of the root servlet you see the start screen as shown in *Figure 3-3*. You should see the root objects *openCRX Administration*, *Codes*, *UOMs*, *Security Realms*, *Security Policies* and *Security Users / Groups*.

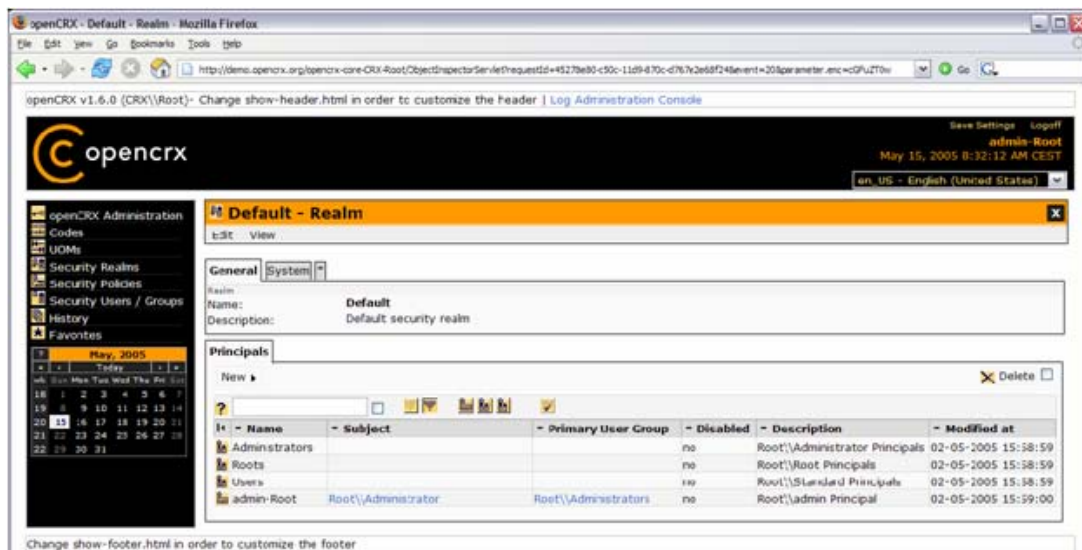
Figure 3-3. Initial screen after login as admin-Root



Important If you started with an empty database (or imported new codes) you **MUST SHUT DOWN** the servlet or your app server, restart it, and then login as *admin-Root* again.

You can now navigate through the *Codes*, *UOMs*, etc. You should see the data which was initially loaded by the servlet. Verify now whether the principal *admin-Root* has been created correctly: click *Security Realms* > select the *Default* realm. You then see a tab containing the principal *admin-Root* and the principal groups **Administrators**, **Roots**, and **Users** as shown in *Figure 3-4*.

Figure 3-4. Realm "Default" is created automatically



Next you must set the access levels of all the codes to **global** (Browse), **basic** (Update), and **basic** (Delete). Open the code provider by clicking on **Codes**. Select the menu *Security > Set Access Level* to unhide the parameter panel and then set the parameters of this operation as shown in *Figure 3-5*.

Figure 3-5. admin-Root sets Access Levels of Codes

The screenshot shows a dialog box titled "Codes" with a menu bar containing "Edit", "View", and "Security". The main area is titled "Set Access Level" and contains an "Input" section with the following fields:

- Browse access level: [4] global (all users)
- Update access level: [2] basic (group members)
- Delete access level: [2] basic (group members)
- Mode: [1] recursive
- Reference filter: (empty text box)

At the bottom of the dialog are "OK" and "Cancel" buttons.

Click the button [OK] to start the operation (please note that this operation can take several minutes on a slower machine...).

Finally, you must set the access levels of all the units of measurement to **global** (Browse), **basic** (Update), and **basic** (Delete). Open the UOM provider by clicking on **UOMs**. Select the menu *Security > Set Access Level* to unhide the parameter panel and then set the parameters of this operation as shown in *Figure 3-6*.

Figure 3-6. admin-Root sets Access Levels of Uoms

The screenshot shows a dialog box titled "UOMs" with a menu bar containing "Edit", "View", and "Security". The main area is titled "Set Access Level" and contains an "Input" section with the following fields:

- Browse access level: [4] global (all users)
- Update access level: [2] basic (group members)
- Delete access level: [2] basic (group members)
- Mode: [1] recursive
- Reference filter: (empty text box)

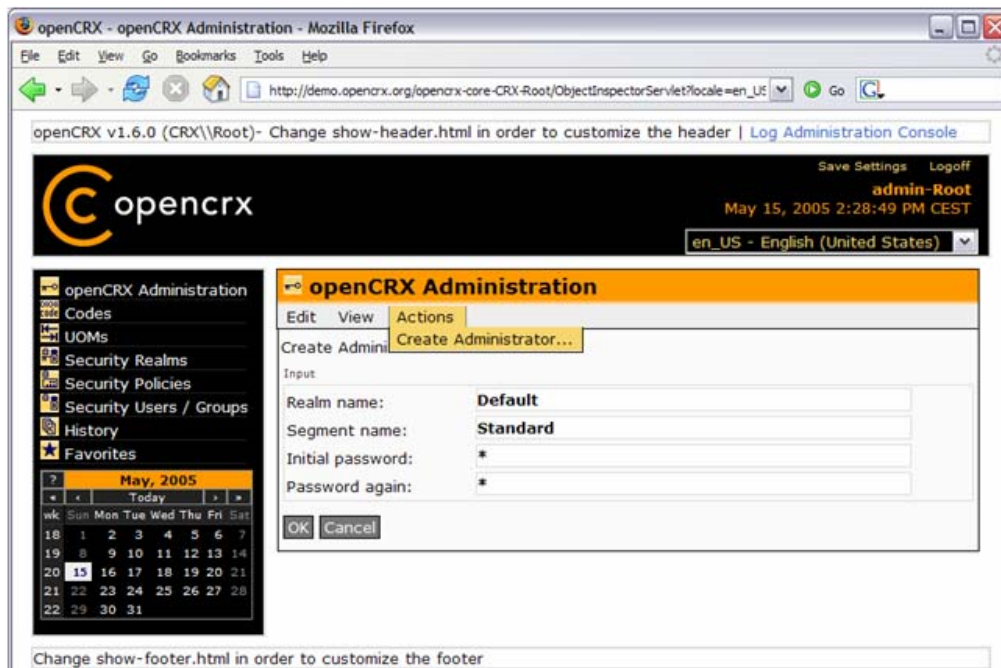
At the bottom of the dialog are "OK" and "Cancel" buttons.

Click the button [OK] to start the operation. Congratulations! You have successfully completed the initial setup of *openCRX*.

Creation of a new data Segment

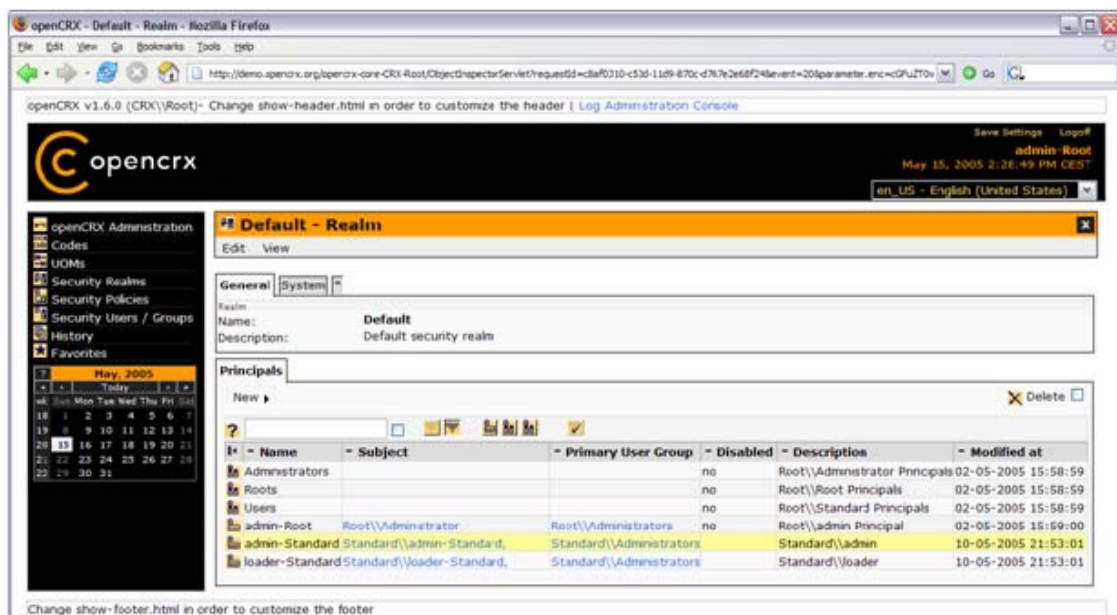
Connect to the **Root GUI** and execute the operation *Actions > Create Administrator* to create a new segment and selected default accounts (including an administrator's account which allows you to manage the newly created segment). Set the field Realm name to *Default* and the Segment name to *Standard*. Set the password to ***. This creates a segment with name *Standard*. This is shown in *Figure 3-7*.

Figure 3-7. Create a new Segment



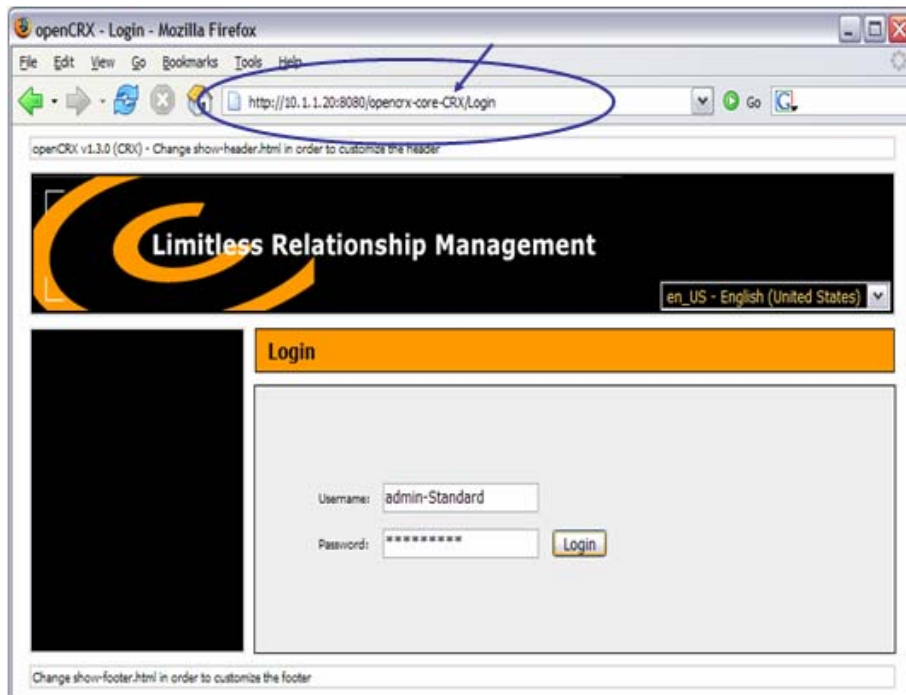
After creating the segment, click on the menu *Security Realms* and navigate to the realm **Default**. Verify whether the principals *admin-Standard* and *loader-Standard* were created as shown in *Figure 3-8*.

Figure 3-8. Verify creation of new principals admin-Standard and loader-Standard



Next you have to open a new browser window and start the **Standard GUI** by entering the URL `http://localhost:8080/openctx-core-CRX/Login`. Login as *admin-Standard* as shown in *Figure 3-9*.

Figure 3-9. Login as admin-Standard



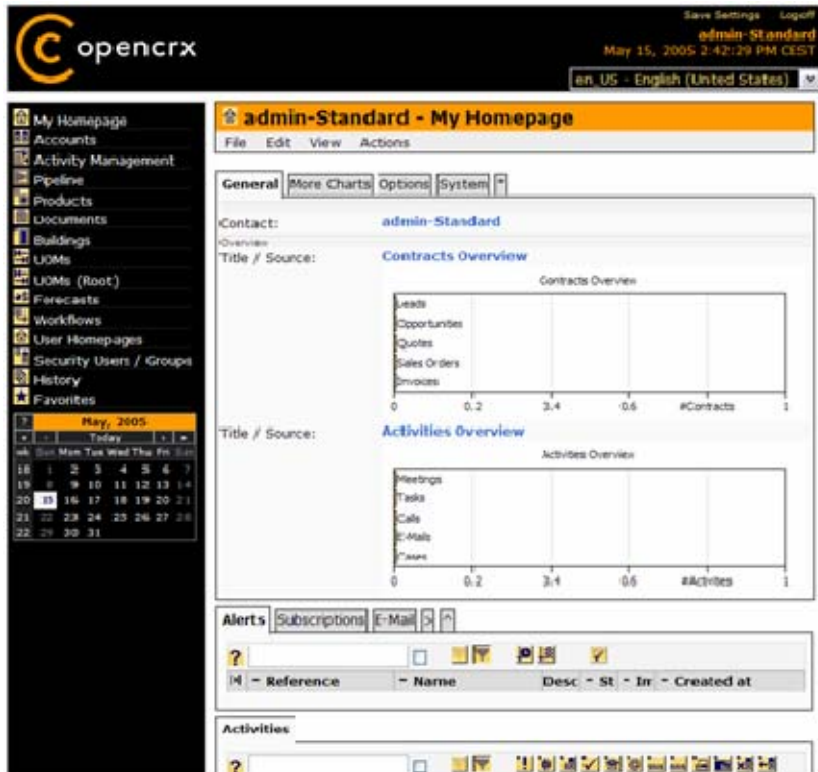
important Before you can login as *admin-Standard* you must add the login to your application server's user list. E.g. with JBoss you must add the user to the file *openCRX.users.properties*:



important After the first successful login as administrator you have to **log out** and **log in** again. This properly initializes the GUI.

Initially, the administrator homepage does not contain any charts. You can activate the charts by executing the operation *View > Recalculate and Refresh* (but do not forget to **log out** and **log in** again if it is the first time you logged in as administrator; otherwise the *Recalculate and Refresh* operation will fail!). Now the administrator homepage should look as shown in *Figure 3-10*.

Figure 3-10. Homepage of admin-standard (charts initialized)

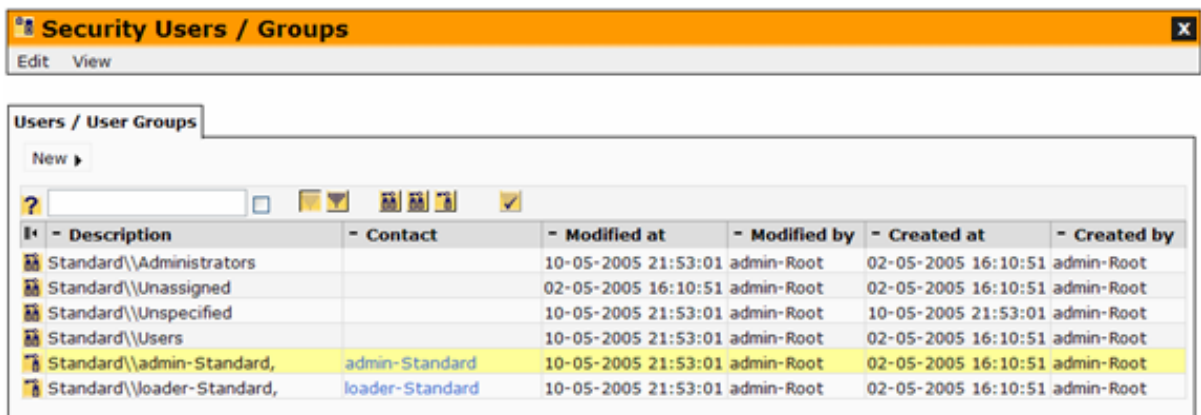


Verify in *Security Users / Groups* whether the following entries exist as shown in *Figure 3-11*:

- *Standard\Administrators*
- *Standard\Unassigned*
- *Standard\Unspecified*
- *Standard\Users*
- *Standard\admin-Standard*
- *Standard\loader-Standard*

These default users and groups were created by the operation *Create Administrator* you previously executed as Root.

Figure 3-11. Verify Users and User Groups



Congratulations! You have successfully created the new segment *Standard*.

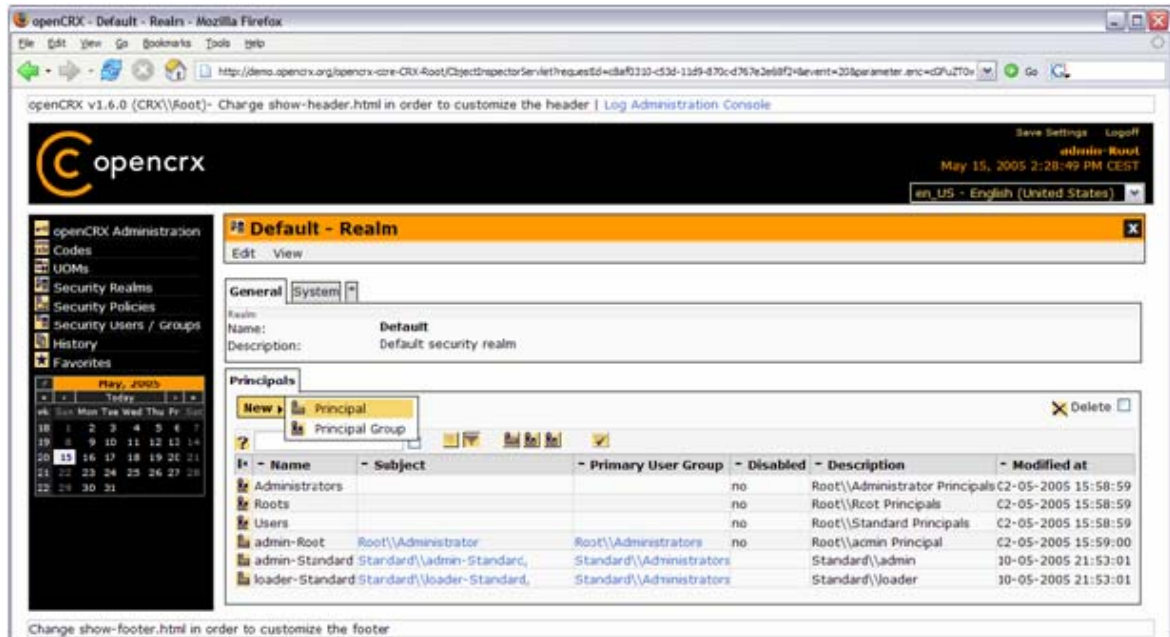
Create a User

The creation of a new user is performed in two steps:

- as Root: create a new principal
- as segment Administrator: create a new user

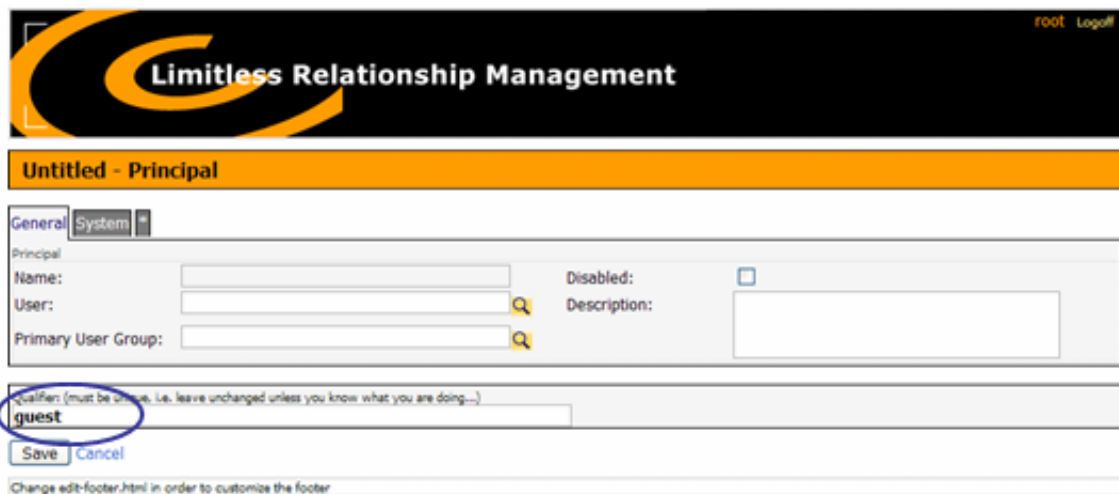
Open a new browser window, start the **Root** GUI by entering the URL `http://localhost:8080/opencrx-core-CRX-Root/Login` and then login as `admin-Root`. Create a new principal by navigating to *Security Realms* > *Default* and then selecting the creator menu *New* > *Principal* as shown in *Figure 3-12*.

Figure 3-12. Create a new User by clicking on the button [Principal]



You only have to set the field qualifier at the bottom of the new principal form. Enter the principal name (login name) of the user, e.g. `guest` as shown in *Figure 3-13*. The other fields can be left blank. They are set automatically by the operation *Create User* which is performed by the segment administrator in the next step.

Figure 3-13. Enter Principal name (=Login name) into the field "Qualifier"

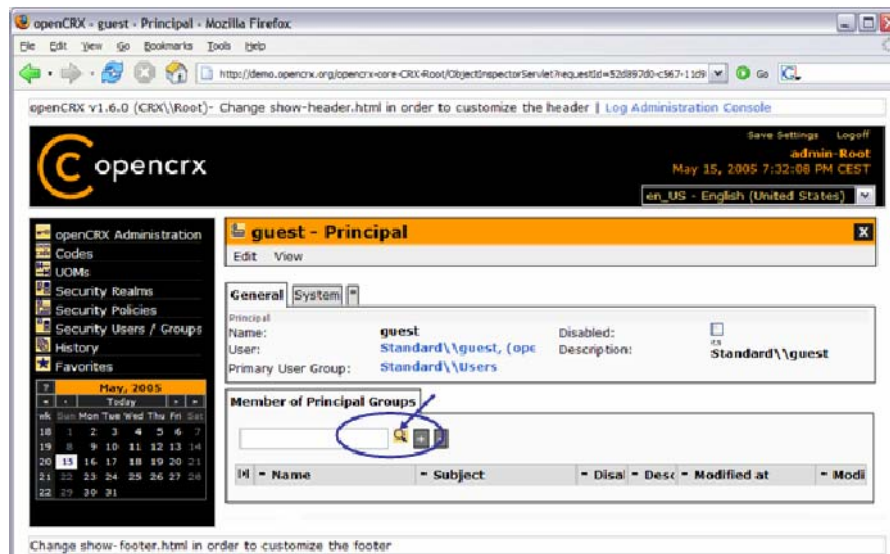




Important Before you can login as *guest* you must add the login to the application server's user list. E.g. with JBoss you must add the login to the file *openCRX.users.properties*.

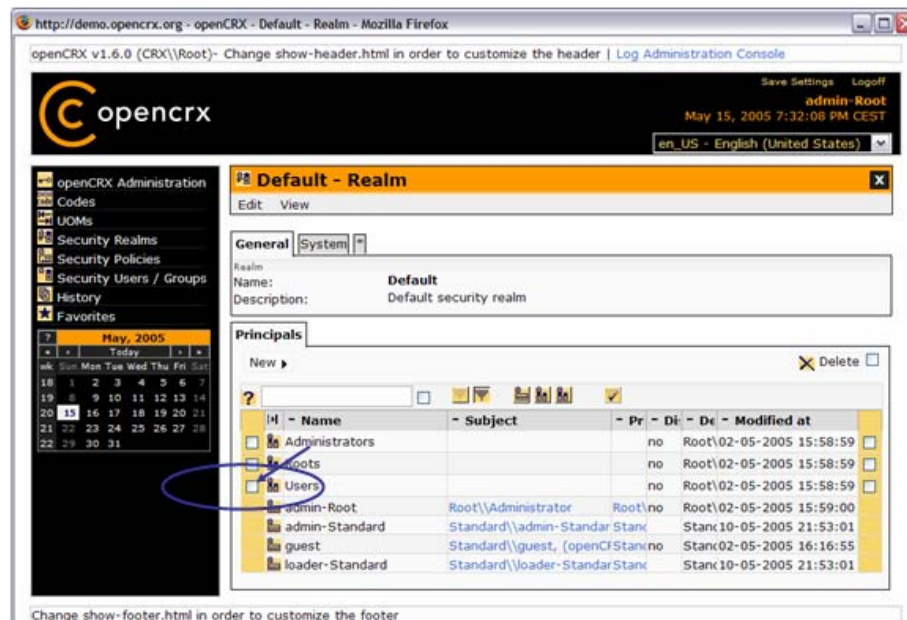
Next you must add the newly created principal to the appropriate principal group. Navigate to the newly created principal (i.e. load it into the inspector) and click on the looking glass in the tab *Member of Principal Groups* to open the Lookup Inspector:

Figure 3-14. Add Principal to Principal Group - Step 1



The Lookup Inspector lists all the principals and principal groups of the respective realm; principal group entries have a check box:

Figure 3-15. Add Principal to Principal Group - Step 2



Click the check box next to the principal group you want to add the newly created principal to (normal users should be added to the principal group *Users*, segment administrators should be added to the principal group *Administrators*). This will automatically close the Lookup Inspector again.



Tip Consult the *openCRX Security Guide* (<http://www.opencrx.org/documents.htm>) for additional/more detailed information on principals and principal groups.

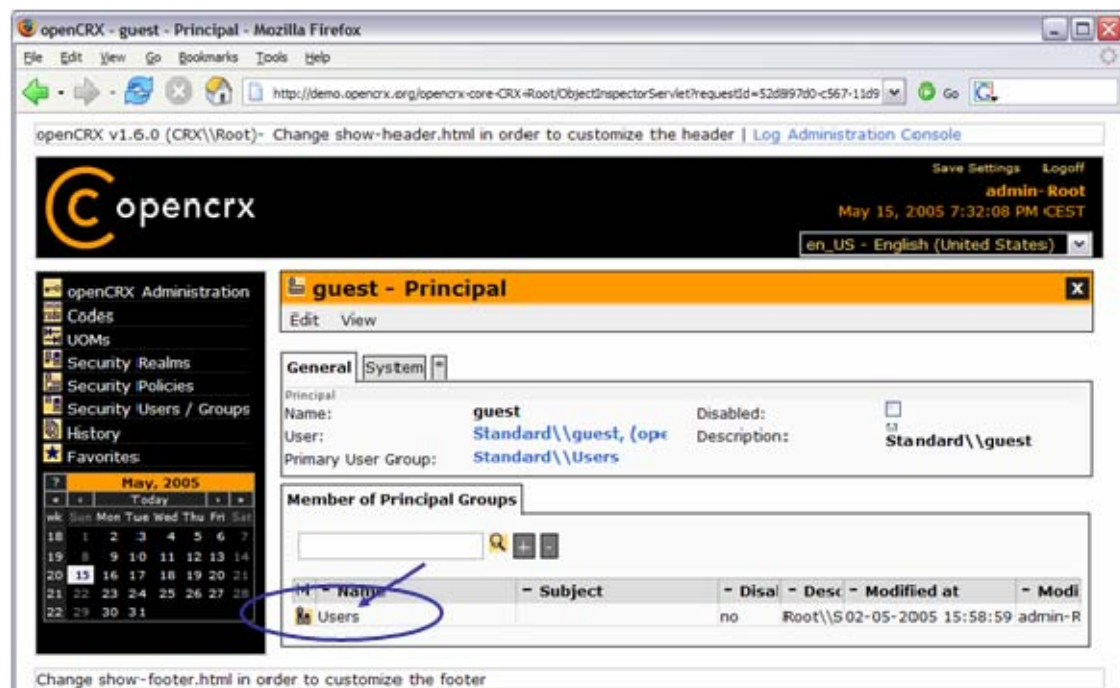
Back in the tab *Member of Principal Groups* you click the button [+] to add the principal to the principal group selected in the previous step.

Figure 3-16. Add Principal to Principal Group - Step 3



Finally, this principal group should show up in the *Grid Member of Principal Groups*:

Figure 3-17. Add Principal to Principal Group - Step 4



Now you have to go back to the **Standard GUI** (the servlet where you are logged in as *admin-Standard*). As administrator you first you have to create a new contact in *Accounts* for the new user *guest*. Enter at least the last name as shown in *Figure 3-18*.

Figure 3-18. As admin-Standard, create a Contact for the new user "guest"

Click the button [Save] to store the contact.

Then click *User Homepages* and select the creator menu *New > My Homepage* which allows you to create and initialize a new user. Set the fields to the values as shown in *Figure 3-19*.

Figure 3-19. Operation [Create User]

| Name | E-mail subject prefix | Web | Stor | User set | Modified at |
|-----------------|-----------------------|-----|------|----------|---------------------|
| admin-Standard | | | | #setting | 10-05-2005 16:04:32 |
| loader-Standard | | | | | 10-05-2005 10:39:30 |

Set the fields *user id* and *principal* name to the login name of the new user, e.g. *guest*. Set the field *realm* to *Default*. Set the field *contact* to the contact you have created previously (use the lookup inspector which appears if you click the looking glass icon) and set the *primary user group* to *Standard\Users* (also use the lookup Inspector to do this). The password fields can be set to the value ***.



Important Passwords are stored in the table *security_Credential*. If you do not have the login module of your application server configured to access the table *security_Credential* then the setting of the passwords in openCRX has no effect.

Click the button [OK] to create the new user. The operation creates a new homepage and links the Principal with the newly created user. You can verify the result of the operation by opening *User Homepages* segment which shows the newly created homepage.



Tip You can usually fix corrupt data of an existing user by creating this user's homepage again – note, however, that the respective user's personal settings are reset to default values.

As *root* you can also verify whether the creation of the new user was successful. In the **Root GUI** Click *Security Realms* and navigate to the realm *Default* where you select the principal *guest*. The fields *user* and *primary user group* should be set as shown in *Figure 3-20*.

Figure 3-20. As root, verify creation of new user "guest"

The screenshot shows the openCRX administration interface in a Mozilla Firefox browser window. The page title is "openCRX - guest - Principal - Mozilla Firefox". The address bar shows the URL: "http://demo.opencrx.org/opencrx-core-CRX-Root/ObjectName/securityServlet?requestId=59285". The page content includes a navigation menu on the left, a main content area, and a footer.

The main content area displays the configuration for the "guest" principal. The "General" tab is selected, showing the following fields:

- Principal Name: **guest**
- User: **Standard\guest**
- Primary User Group: **Standard\Users**
- Disabled:
- Description: **Standard\guest**

Below the "General" tab, there is a section for "Member of Principal Groups" with a search input field and a list of groups. The list shows the following data:

| Name | Subject | Dis | De | Modified at | Mo |
|-------|---------|-----|--------------------------|-------------|----|
| Users | | nc | Root\02-05-2005 15:58:59 | admin | |

The footer of the page contains the text: "Change show-footer.html in order to customize the footer".

Congratulations! You have successfully created a new user *guest*. The new user should now be able to login in the **standard GUI**.

Once the new user opens a browser window and connects to the **standard GUI** by entering the URL `http://localhost:8080/opencrx-core-CRX/Login` he will be able to login as *guest* as shown in *Figure 3-21*.

Figure 3-21. The new user "guest" can login in the standard GUI



enlarge (./images/Create-User-6_enlarged.png.htm)



Important The new user login must be added to the application server's user list, e.g. with JBoss to the file `openCRX.users.properties`:



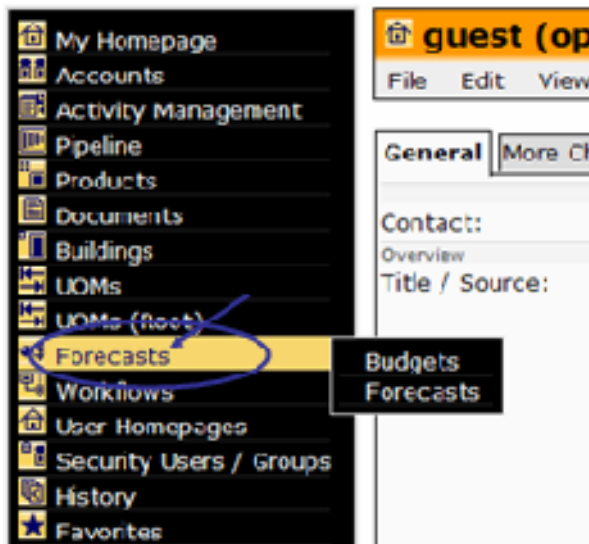
Important After the first successful login as *guest* you have to **log out** and **log in** again. This properly initializes the GUI.

Chapter 4. Managing the Provider List

The default installation of *openCRX* activates all providers that are included in the Open Source distribution. The *openCRX* administrator may wish to remove certain providers from the provider list. This chapter shows how you can achieve this.

Let us assume that the *openCRX* administrator decided to hide the provider **Forecasts**.

Figure 4-1. Provider List



The provider list is contained in the file *web.xml* which is contained in the file *openCRX-core-CRX.war* (contained in *openCRX-core-CRX-web.ear*, both files are archive files that can be opened with a ZIP utility) deployed to your application server. The following discussion assumes that you use the **JBoss** (<http://www.jboss.org/>) application server.



Important In the case of the JBoss application server you should find the file *openCRX-core-CRX-web.ear* in the directory `.../jboss-4.0.1/server/default/deploy`. To make changes to files contained in the file *openCRX-core-CRX-web.ear* you can basically decide between the following two approaches (remember to stop the application server **BEFORE** you make any changes!):

- (1) expand/unzip the *EAR*, (2) extract the file to be changed from the *WAR*, (3) change the file, (4) add the changed file back to the *WAR*, and then (5) pack/zip the *EAR* back together; this process is obviously pretty cumbersome if you intend to make frequent changes
- you can first extract all the files contained in the *EAR* file to a directory with the same name as the *EAR* file, e.g. *openCRX-core-CRX-web.ear*; you can then extract all the files contained in the *WAR* file to a directory with the same name as the *WAR* file, e.g. *openCRX-core-CRX.war* (to avoid naming conflicts you do this by extracting the files to a temporary directory, delete the *WAR* file, and then rename the temporary directory to *openCRX-core-CRX.war*); now you can change files without unzipping/zipping

Let us now proceed. First you need to stop your application server. Then you locate the file *web.xml* in the directory `.../jboss-4.0.1/server/default/deploy/openCRX-core-CRX-web.ear/openCRX-core-CRX.war/WEB-INF`. Open it with an editor and locate the following few lines:

Example 4-1. web.xml with the provider declaration for Forecasts

```

<!-- Forecasts -->
<init-param>
  <param-name>rootObject[9]</param-name>
  <param-value>xri:@openmdx:org.opencrx.kernel.forecast1/provider/CRX/segment/${SEGMENT}</param-value>
</init-param>
<init-param>
  <param-name>rootObjectClass[9]</param-name>
  <param-value>org.opencrx:kernel:forecast1:Segment</param-value>
</init-param>

```

To hide this provider you can either delete the above lines or comment them out as shown in the following Example:

Example 4-2. web.xml with the Forecasts provider commented out

```

<!-- Forecasts -->
<!--
<init-param>
  <param-name>rootObject[9]</param-name>
  <param-value>xri:@openmdx:org.opencrx.kernel.forecast1/provider/CRX/segment/${SEGMENT}</param-value>
</init-param>
<init-param>
  <param-name>rootObjectClass[9]</param-name>
  <param-value>org.opencrx:kernel:forecast1:Segment</param-value>
</init-param>
-->

```



Important Providers must be numbered with consecutive numbers (i.e. you must not skip numbers!). If you for example decided to hide the Forecasts provider then you **must** adapt the numbers of the following providers (Workflows, User Homepages, Security/Identity) accordingly.

Save the changed file and restart your application server. The provider list of *openCRX* will not show “Forecasts” anymore.

To unhide/reactivate this provider, simply uncomment the declarations in the file *web.xml*.



Tip With some application servers it is necessary to delete temporary directories and working directories before starting the application server again to make sure that the configuration changes are properly processed.

Chapter 5. Security



Warning We do not recommend learning about security with mission critical data. Backup your data **BEFORE** you make changes if you are not certain what the consequences are! The default settings should work for virtually all users; the probability of getting yourself into troubles by changing default settings should not be underestimated. Read and understand the *openCRX Security Guide* (<http://www.opencrx.org/documents.htm>) **BEFORE** you make any changes.

Starting with openCRX v1.4.0 access control is activated. Please refer to the *openCRX Security Guide* (<http://www.opencrx.org/documents.htm>) for a detailed explanation of role-based security as it is implemented by openCRX. In this chapter we discuss a few select issues only, including disaster recovery if you locked yourself out and/or screwed up the security settings in a major way.

Security Setting of New Objects

New objects are by default created with the following security settings:

- Owing User: **User who is creating the object**
- Browse Access Level: **3 – deep**
- Update Access Level: **2 – basic**
- Delete Level: **2 – basic**
- Owing Groups: **Primary User Group of the user who is creating the object**



Important Please note that a users's **primary user group** is set by the segment administrator with the operation [Create User]. To change an existing user's primary group, the segment administrator simply executes the operation [Create User] again with a new parameter for primary user group.



Tip In the context of activity management there are various operations that set/change the Owing Groups of objects based on the settings of an assigned Activity Tracker and not based on the settings of the user who executes the operation.

If you see **N/P** in a field instead of a more meaningful value you probably do not have browse access to the respective object (**N/P** stands for **No Permission**).

Resetting Security



Important If you get the setting of **Update Access Level** wrong you may not be able to change the respective object from the GUI anymore. For example, the only way to recover from setting Update Access Level to **N/A** for a particular object is to edit the data directly in the database!

If you (or one of your users) managed to screw up the security settings in a major way you may be forced to reset all security settings to a well-defined state. Included in the openCRX distribution are example scripts **set-access-level.sql** for all supported DBs. Please refer to the *openCRX Security Guide* (<http://www.opencrx.org/documents.htm>) for a detailed explanation of security **BEFORE** you execute any scripts. **Consider running the complete script *set-access-level.sql* as a last resort only!** In many cases it you will probably get away with fixing the security settings of a particular object on the DB.

Chapter 6. Importing Data

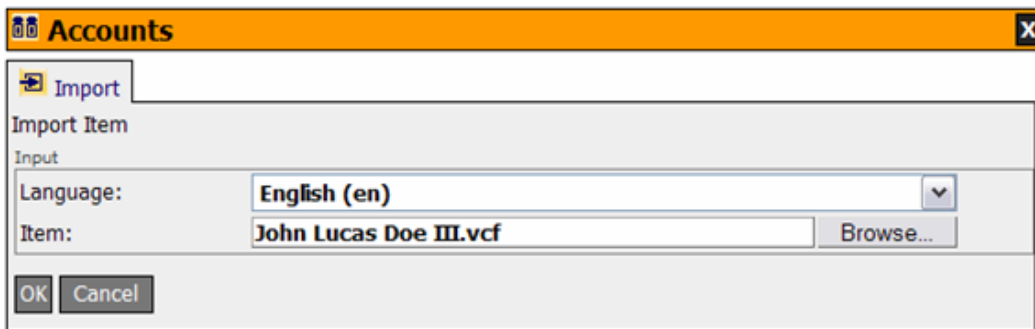
The task of importing data is handled by importers. In principle, you can import almost anything into *openCRX*, it's really only a matter of writing an importer. The Open Source distribution of *openCRX* includes importers for vCard and iCalendar files in addition to the XML importer. This allows you to import data from a large variety of other programs, including Microsoft Outlook. This chapter shows how to import data.

Importing vCard Files

These are the steps to import a vCard file:

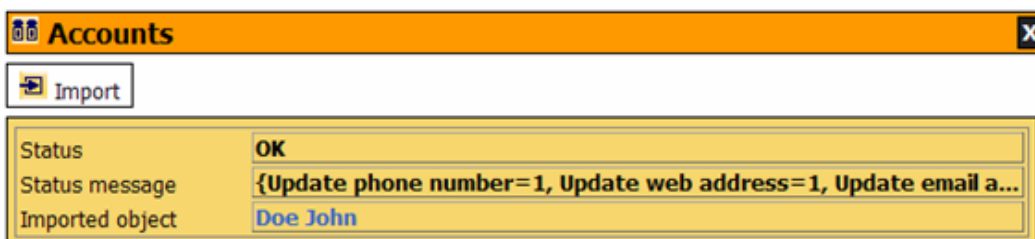
- click on the provider *Accounts*
- click on the operations button [Import] to unhide the import dialog
- select the appropriate language
- click the Browse button and navigate to the vCard file you want to import
- click to button [OK] to start the import operation

Figure 6-1. Operation vCard import



The result of the import operation also contains a link to the created/updated contact (only if the import was successful).

Figure 6-2. Result of the vCard import operation

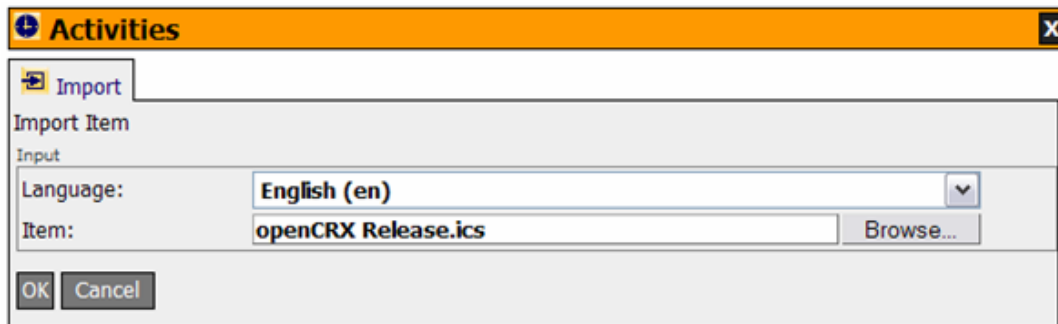


Importing iCalendar Files

These are the steps to import an iCalendar file:

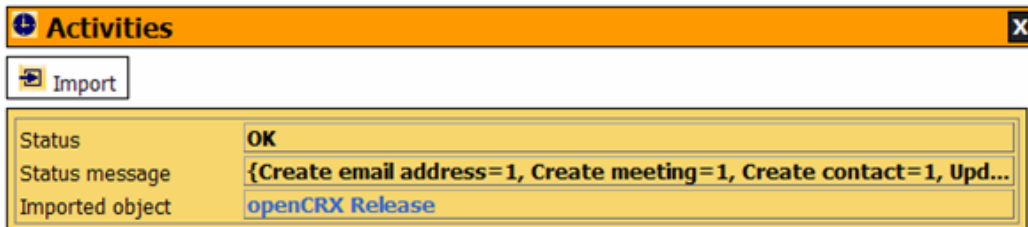
- click on the provider *Activities*
- click on the operations button [Import] to unhide the import dialog
- select the appropriate language
- click the Browse button and navigate to the iCalendar file you want to import
- click to button [OK] to start the import operation

Figure 6-3. Operation iCalendar import



The result of the import operation also contains a link to the created/updated meeting (only if the import was successful).

Figure 6-4. Result of the iCalendar import operation

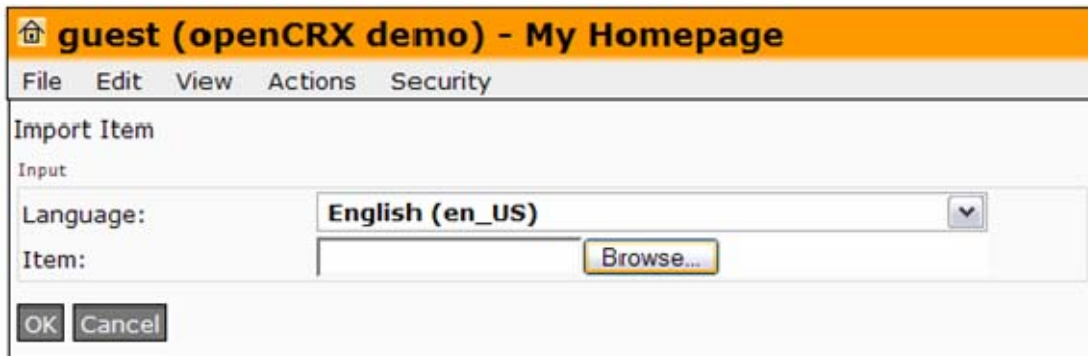


Importing XML Files

These are the steps to import an XML file:

- click on the provider *My Homepage*
- click on the operations button [Import] to unhide the import dialog
- select the appropriate language
- click the Browse button and navigate to the XML file you want to import
- click to button [OK] to start the import operation

Figure 6-5. Operation Import on UserHome



Please note that only schema-compliant XML files can be imported.

Chapter 7. Exporting Data

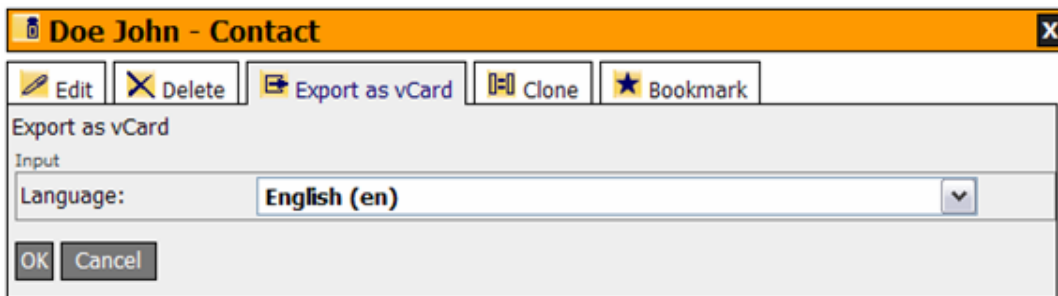
The task of exporting data is handled by exporters. In principle, you can export any object from *openCRX*, it's really only a matter of writing an exporter. The Open Source distribution of *openCRX* includes exporters for vCard and iCalendar files in addition to XML files. This allows you to export contacts and meetings/sales visits or any other object from *openCRX*. vCard and iCalendar files can be imported by a large variety of other applications, including Microsoft Outlook. This chapter shows how to export data.

Exporting vCard Files

These are the steps to export a contact to a vCard file:

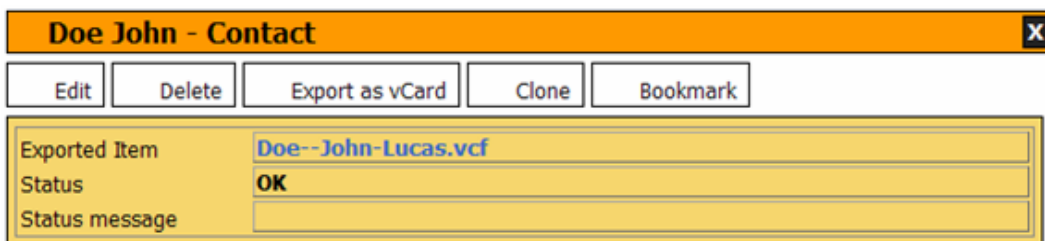
- click on the provider *Accounts* and navigate to the contact you want to export
- click on the operations button [Export as vCard] to unhide the export dialog
- select the appropriate language
- click to button [OK] to start the export operation

Figure 7-1. Operation vCard export



If the export operation was successful the result will contain a link to the vCard file. Click on that link to download the vCard file from the *openCRX* server.

Figure 7-2. Result of the vCard export operation

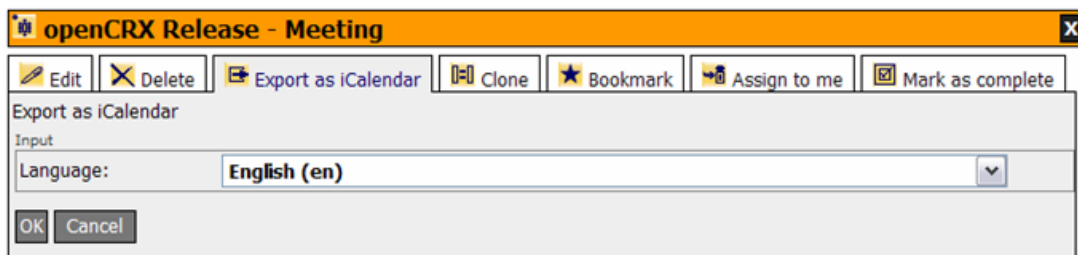


Exporting iCalendar Files

These are the steps to export a meeting (or a sales visit) to an iCalendar file:

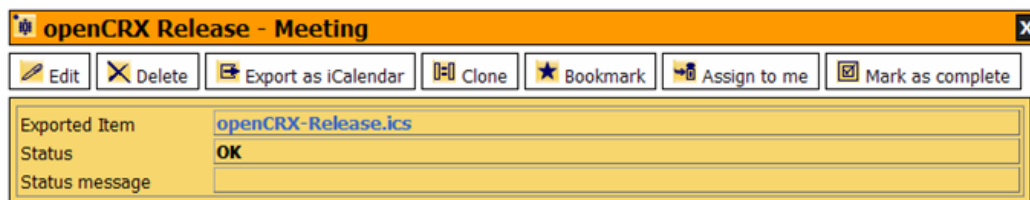
- click on the provider *Activities* and navigate to the meeting (or sales visit) you want to export
- click on the operations button [Export as iCalendar] to unhide the export dialog
- select the appropriate language
- click to button [OK] to start the export operation

Figure 7-3. Operation iCalendar export



If the export operation was successful the result will contain a link to the iCalendar file. Click on that link to download the iCalendar file from the *openCRX* server.

Figure 7-4. Result of the iCalendar export operation

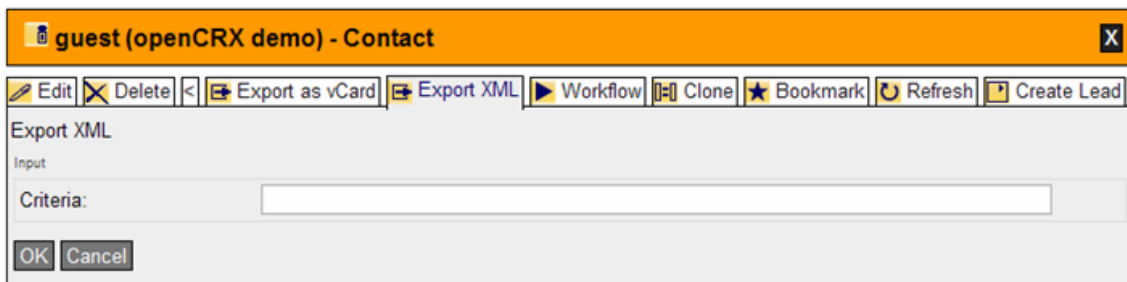


Exporting XML Files

All objects can be exported as XML files. These are the steps to export (for example) a meeting (or a sales visit) to an XML file:

- click on the provider *Activities* and navigate to the meeting (or sales visit) you want to export
- click on the expand button [>] and then on the operations button [Export XML] to unhide the export dialog
- optional: enter the desired level of dereferencing into the field **Criteria** – if you leave this field empty a default value of "1" is used (this is usually sufficient)
- click to button [OK] to start the export operation

Figure 7-5. Operation Export XML

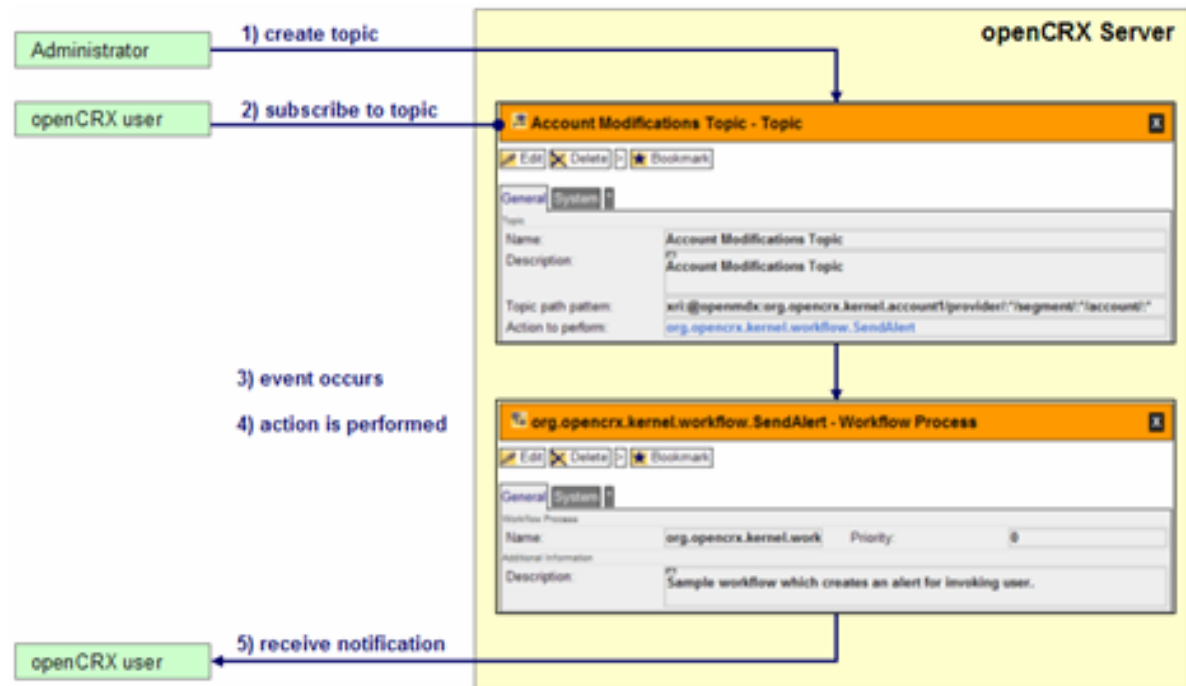


The XML file is automatically zipped. You can download this file and save it.

Chapter 8. The Event Subscription / Notification Service

openCRX features a powerful event subscription and notification service (see *Figure 8-1* for an overview):

Figure 8-1. Overview Event Subscription / Notification



Administrators can create **Topics** containing information like a "topic path pattern" and "action to perform". The *openCRX* distribution includes quite a few default topics (see *Figure 8-2*) to get you started, e.g.

- *Account Modification* sends an alert to the UserHome of subscribed users if any account is modified
- *Alert Modification* sends an e-mail alert to subscribed users if any of the alerts on their UserHome is modified
- *Lead Modification* sends an alert to the UserHome of subscribed users if any lead is modified
- ...

Once a topic is created *openCRX* users can subscribe to it. Users manage their subscriptions individually on their UserHomes. If a topic has subscribed users and a monitored event occurs then the predefined action is performed. If the action is set to – for example – creating an alert for subscribed users, then each subscribed user will receive an alert on his UserHome.

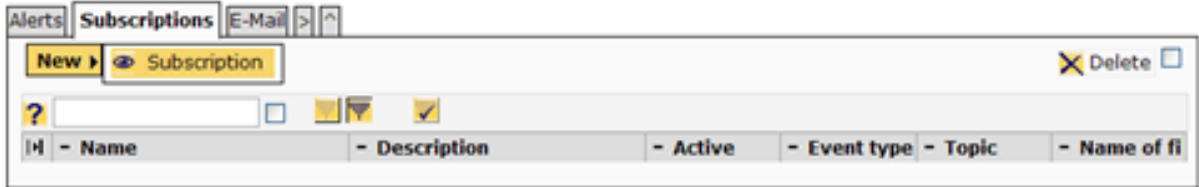
Figure 8-2. Topics included in the openCRX distribution

| Name | Description | Topic path pattern | Action to perform | Modified at | Modified by |
|---------------------------------|---------------------------------|---|-------------------------------------|---------------------|-----------------|
| Activity Modifications | Activity Modifications | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/activity):* | org.openrx.kamel.workflow.SendAlert | 09-05-2005 03:27:33 | wfro |
| Account Modifications Topic | Account Modifications Topic | xri:@openmdc.org:openrx.kamel.account1(provider:*/segment:*/account):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Meeting Modifications Topic | Meeting Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/activity):* | org.openrx.kamel.workflow.SendAlert | 09-05-2005 02:22:36 | leader-Standard |
| Alert Modifications Topic | Alert Modifications Topic | xri:@openmdc.org:openrx.kamel.home1(provider:*/segment:*/userHome:*/alert):* | org.openrx.kamel.workflow.SendMail | 15-02-2005 20:52:58 | wfro |
| Calls Modifications Topic | Calls Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/phoneActivity):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:23 | admin-Standard |
| Case Modifications Topic | Case Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/incident):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Competitor Modifications Topic | Competitor Modifications Topic | xri:@openmdc.org:openrx.kamel.account1(provider:*/segment:*/competitor):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Email Modifications Topic | Email Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/emailActivity):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:23 | admin-Standard |
| Fax Modifications Topic | Fax Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/faxActivity):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Invoice Modifications Topic | Invoice Modifications Topic | xri:@openmdc.org:openrx.kamel.contract1(provider:*/segment:*/invoice):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Lead Modifications Topic | Lead Modifications Topic | xri:@openmdc.org:openrx.kamel.contract1(provider:*/segment:*/lead):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Mailing Modifications Topic | Mailing Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/mailingActivity):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Meeting Modifications Topic | Meeting Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/meeting):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:23 | admin-Standard |
| News Modifications Topic | News Modifications Topic | xri:@openmdc.org:openrx.kamel.activity1(provider:*/segment:*/newsActivity):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |
| Opportunity Modifications Topic | Opportunity Modifications Topic | xri:@openmdc.org:openrx.kamel.contract1(provider:*/segment:*/opportunity):* | org.openrx.kamel.workflow.SendAlert | 15-02-2005 20:22:26 | admin-Standard |

Example Subscription – New Accounts

In this example we will create a subscription to the standard topic *Account Modifications Topic* so that we will receive an alert on our UserHome whenever a new account is created. Login to *openCRX* (or navigate to your UserHome if you are already logged in). Click on the Grid tab *Subscriptions* to see all your current subscriptions:

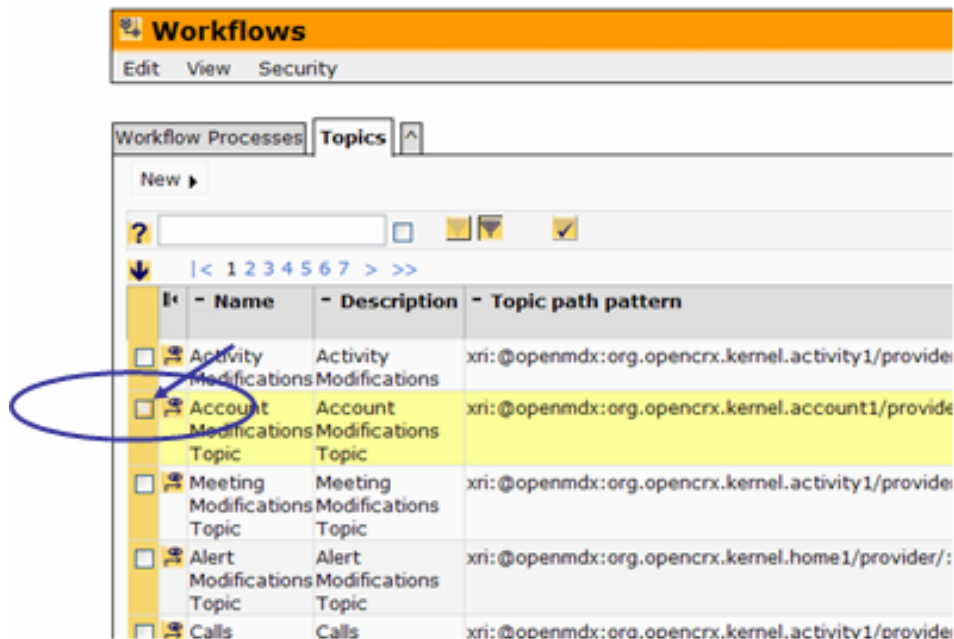
Figure 8-3. Create a new subscription



Click on the creator menu *New > Subscription* to create a new subscription.

Enter a *name* and a *description* of the subscription to be created and check the box *Active*. Click on the edit icon of the field *Event type* and then select the option **[1] Object Creation**; this will ensure that we will get alerts for new accounts only (if you leave the field *Event type* empty you will get alerts for all event types, i.e. Object Creation, Object Modification, and Object Removal). Then you click the lookup icon next to the field *Topic* to start the lookup inspector where you select the topic entry *Account Modifications Topic*:

Figure 8-4. Lookup Inspector for Topics



Your new subscription should now look as follows:

Figure 8-5. Subscription "New Account"

To complete the process you click the button [Save].

Back on your UserHome you should now see an entry for your newly created subscription:

Figure 8-6. UserHome with the newly created subscription "New Account"

| Name | Description | Active | Event type | Topic | Name of filter #0 |
|-------------|--------------------------|--------|---------------------|-----------------------------|-------------------|
| New Account | New Account Subscription | yes | [1] Object Creation | Account Modifications Topic | |

You can test your new subscription with the following steps:

- create a new account (e.g. a new contact)
- navigate to your homepage and check whether you actually received an alert related to the newly created account



Tip In many cases you can make use of the operations *Add Subscription*, *Remove Subscription*, *Add Subscription for Parent*, and *Remove Subscription for Parent* located in the operations menu *Tools*. If your segment administrator has created the respective Topics these operations are a quick way to create/delete subscriptions. Try this out by navigating to a contact and executing the operation *Tools > Add Subscription for Parent* – this will add the subscription **Account Modifications Topic** to your list of subscriptions.

Chapter 9. Outbound E-mail Service

openCRX users can configure an E-mail account on their UserHome indicating where they would like to receive e-mail notifications (e.g. generated by a subscription to the topic "Alert Modifications Topic"). Click on *My Homepage* and select the Grid Tab **[E-Mail]**. Next you click on the creator menu *New > E-Mail Account* to create a new E-mail Account:

Figure 9-1. Create a new E-mail Account



Now you can define your E-mail Account for outbound e-mail service (SMTP):

Figure 9-2. E-mail Account Object

The screenshot shows a configuration form titled 'Untitled - E-Mail Account'. It has two tabs: 'General' and 'System'. The 'General' tab is active. The form is divided into 'User Information' and 'Outgoing Mail Server (SMTP)' sections. Under 'User Information', there are fields for 'E-mail address' (user@mycompany.com), 'Reply address' (noreply@mycompany.com), and a 'Default' checkbox (checked). Under 'Outgoing Mail Server (SMTP)', there are fields for 'Mail server (SMTP)' (smtp.mycompany.com), 'Mail server port' (25), 'User name' (user), 'Password' (masked with asterisks), and 'Authentication required' (checked). There is also an 'SSL/TLS' checkbox (checked). At the bottom, there is a 'Qualifier' field with the value 'e9273af0-c578-11d9-bd57-ef2cebacc7238' and 'Save' and 'Cancel' buttons.

- **E-mail address:** enter your e-mail address (i.e. the address where you would like to receive e-mail notifications)
- **Reply address:** enter a reply address (is also used for the **From** field)
- **Default:** check if this is your default e-mail address (notifications will only be sent to your default e-mail address)
- (Outgoing) **Mail server (SMTP):** openCRX will connect to this SMTP server to submit e-mail notifications
- (Outgoing) **Mail server port:** the default port for SMTP is 25, but you can change the port if your mail server is listening on a different port
- (Outgoing) **User name:** enter the user name to be used for authentication purposes (only used if *Authentication required* is checked)
- (Outgoing) **Password:** enter the password to be used for authentication purposes (only used if *Authentication required* is checked)
- (Outgoing) **Authentication required:** if checked, *openCRX* will authenticate with *User name* and *Password* to get access to the mail server
- (Outgoing) **SSL/TLS:** check this option if the outgoing mail server supports TLS and you want to encrypt the communication between *openCRX* and the mail server

Click the button **[Save]** to complete the creation of your E-mail Account.

Back on *My Homepage* you should see an entry in the Grid Tab [**E-Mail**] reflecting your new E-mail Account:

Figure 9-3. Grid Tab [E-mail]



If you edit your Homepage you can set additional E-mail options which are valid system-wide for your *openCRX* account:

Figure 9-4. E-mail Options



- **E-mail subject prefix:** enter a string that might help you identifying or filtering e-mails from your *openCRX* server (optional, i.e. you can also leave this empty) – the entered string is prepended to the subject line of generated e-mails
- **Web access URL:** enter the URL of the *openCRX* instance at hand; if entered correctly, generated e-mails will contain URLs that allow you to connect to your *openCRX* server with a single click

Click the button [**Save**] to update your Homepage..

You can test your new subscription with the following steps:

- create a new account (e.g. a new contact)
- navigate to your homepage and check whether you actually received an alert related to the newly created account
- next click on the Grid Tab [Workflow Process Instances] on your homepage (unhide it by clicking on [>] if it is not visible)
- there should be an entry **org.opencrx.kernel.workflow.SendMail** (you might have to sort the column Started on to located the recent entries)
- click on the icon of the respective grid entry to inspect the corresponding **Workflow Process Entry** object
- the grid **Action Log Entries** contains either the message body of the e-mail that was sent or an error message if the workflow failed (please note that it is not unusual to see a "timeout" error message even if you actually received an e-mail; the reason is an e-mail server with high latency – try sending out notifications through an e-mail server that is responsive).



Important If the mail server is not very responsive you might get warnings about failed deliveries from *openCRX* to the mail server even though respective e-mails are actually delivered to you. One of the solutions to this problem is the use of a more responsive mail server.

Default Mail Server

As a segment administrator you can define a default mail server. This allows users of your segment to send outbound e-mails (e.g. notification e-mails) through this default mail server without configuring their own mail server. Your segment users do not need to know any access information nor do they need any login credentials. Here are the steps to configure a default mail server for your segment:

- connect to the **standard GUI** and login as segment administrator (e.g. *admin-Standard*)
- click on *My Homepage* and select the Grid Tab [**E-Mail**]. Next you click on the creator menu *New > E-Mail Account* to create a new E-mail Account
- provide all the relevant information (including credentials required to send e-mail – your credentials are safe as they are not accessible by other segment users)
- replace the generated qualifier with the string **SYSTEM** as shown in *Figure 9-5* and then click the button [Save] to create this new e-mail account

Figure 9-5. Create a Default Mail Server E-mail Account

The screenshot shows a configuration window titled "Untitled - E-Mail Account" with two tabs: "General" and "System". The "General" tab is selected. The form contains the following fields and values:

- User Information:**
 - E-mail address:
 - Reply address:
- Outgoing Mail Server (SMTP):**
 - Mail server (SMTP):
 - Mail server port:
 - User name:
 - Password:
 - Authentication require:
 - SSL/TLS:
- Qualifier (might be unique, i.e. leave unchanged unless you know what you are doing...):**
 -

At the bottom of the window are "Save" and "Cancel" buttons. A blue circle highlights the "SYSTEM" text in the Qualifier field, and a blue arrow points to it from the left.

Tell your segment users that they do not need to provide any information regarding *Outgoing Mail Server (SMTP)* (i.e. they can leave the respective fields empty). Furthermore, **segment users must uncheck the option Default** on their e-mail account settings to indicate that e-mails should be sent through the default mail server as configured by the segment administrator (but segment users must still provide a valid e-mail address and a reply address).

Chapter 10. Integration of MS Word

The basic approach we take to merge *openCRX* data with your MS Word templates is as follows (a detailed example is available *on the openCRX website* (<http://www.opencrx.org/opencrx/1.5/new.htm#mswordxmlmerge>)): create XML export of the object at hand (including referenced objects and all the required code tables), load a document based on a prepared template with MS Word and then pass the XML files to that document where VBA code will merge the data into the document. The following steps are required to configure this feature (it is expected that the Administrator takes care of this, i.e. we do not expect end users to go through the following steps...):

- create a template with placeholders of the sort <<tagName>>; you can either start with a template included in the *openCRX* distribution (templates are located in *opencrx-core-CRX-web.ear\opencrx-core-CRX.war\documents*), download the *MS Word Template "sample-letter"* (<http://www.opencrx.org/downloads/sample-letter.zip>) that already contains lots of VBA helper functions or you can start from scratch and write your own library
- adapt the VBA code specific to the template (i.e. you need to make sure that all your placeholders are replaced with openCRX data when the code is executed)_

Example 10-1. Sample VBA code (sample-letter.doc)

```
contact = getObj(xmlclean, "org.opencrx.kernel.account1.Contact")
localeIdx = 0
Call ReplaceField("salutationCode$ShortText", getCodeValueText(xmlclean, "salutationCode",
Val(getTagValue(contact, "salutationCode")), localeIdx, False), False)
Call ReplaceField("lastName", getTagValue(contact, "lastName"), False)
'primary mailing address
ReDim usagefilter(0, 1)
usagefilter(0, 0) = "usage"
usagefilter(0, 1) = "300" 'primary
mailingAddress = getObjList(getContent(xmlclean, "org.opencrx.kernel.account1.Contact"),
"address", usagefilter, "postalStreet")
Call ReplaceField("postalAddressLine", getTagMultiValueAsS-
tring(getTagValue(mailingAddress(0), "postalAddressLine"), itemTag), False)
Call ReplaceField("postalStreet", getTagMultiValueAsString(getTagValue(mailingAddress(0),
"postalStreet"), itemTag), False)
Call ReplaceField("postalCity", getTagValue(mailingAddress(0), "postalCity"), False)
Call ReplaceField("postalState", getTagValue(mailingAddress(0), "postalState"), False)
Call ReplaceField("postalCode", getTagValue(mailingAddress(0), "postalCode"), False)
Call ReplaceField("postalCountry$ShortText", getCodeValueText(xmlclean, "country",
Val(getTagValue(mailingAddress(0), "postalCountry")), localeIdx, False), False)
phoneNumber = getObjList(getContent(xmlclean, "org.opencrx.kernel.account1.Contact"), "ad-
dress", usagefilter, "phoneNumberFull")
Call ReplaceField("primaryPhone", getTagValue(phoneNumber(0), "phoneNumberFull"), False)
```

- store your template in a place where it can be read by all those users who need to create documents based on it (e.g. on your WebDAV server, on your file server, or directly on the openCRX server, e.g. in the directory *opencrx-core-CRX-web.ear\opencrx-core-CRX.war\documents* like the sample templates)
- add an <additionalElementDefinition> block to the XML exporter section of the ui configuration file *common.xml* and set the template attribute so that it points to your template_

Example 10-2. additionalElementDefinition for MS Word Sample Letter

```
<org.openmdx.ui1.ElementDefinition
name="org:opencrx:kernel:base:XmlExporter:Pane:Op:Tab:exportXml">
  <_object>
    <active>true</active>
    <toolTip>
      <_item>Export XML</_item>
    </toolTip>
    <label>
      <_item>>Export XML</_item>
    </label>
```

```

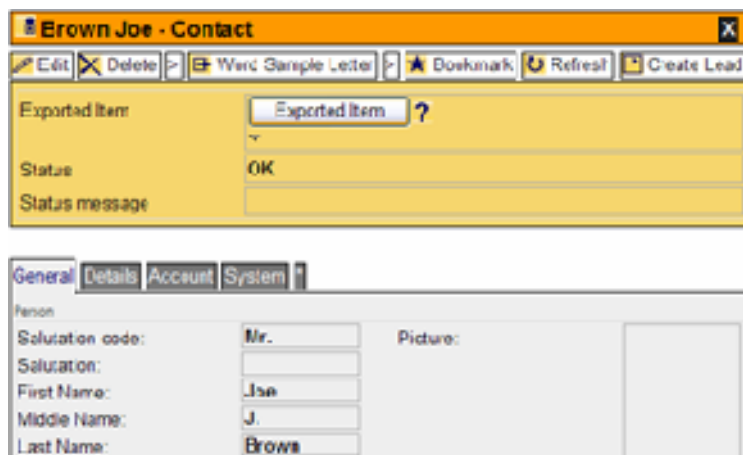
<iconKey>org:opencrx:kernel:Export</iconKey>
<order>
  <_item>0</_item>
  <_item>0</_item>
  <_item>5</_item>
</order>
</_object>
<_content>

<additionalElementDefinition>
  <org.openmdx.ui1.AdditionalElementDefinition id="ExportContactToWord">
    <_object>
      <active>true</active>
      <toolTip>
        <_item>Export to Word</_item>
      </toolTip>
      <label>
        <_item>Word Sample Letter</_item>
      </label>
      <iconKey>org:opencrx:kernel:Export</iconKey>
      <order>
        <_item>0</_item>
        <_item>0</_item>
        <_item>6</_item>
      </order>
      <forClass>
        <_item>org:opencrx:kernel:account1:Contact</_item>
      </forClass>
      <memberDefinitionElementName>
        <_item>item</_item>
      </memberDefinitionElementName>
      <memberDefinitionMimeType>
        <_item>application/x-openmdx-xml-for-word-export;template=documents/sample-
letter.doc;macro=xmlMerge;visible=true</_item>
      </memberDefinitionMimeType>
    </_object>
  </org.openmdx.ui1.AdditionalElementDefinition>
</additionalElementDefinition>
</_content>
</org.openmdx.ui1.ElementDefinition>

```

- after restarting your application server you can create merged Word documents with a single click

Figure 10-1. Ready to launch MS Word



It is probably helpful if you look at the sample that is provided with the distribution (details are published *on the openCRX website* (<http://www.opencrx.org/opencrx/1.5/new.htm#mswordxmlmerge>)).

Chapter 11. Next Steps

Appendix A. Appendix

Bibliography

[01] *openCRX - the leading open source CRM solution*, [opencrx.org](http://www.opencrx.org).

@ <http://www.opencrx.org> (<http://www.opencrx.org>)

[02] *openMDX - The leading open source MDA platform*, [openmdx.org](http://www.openmdx.org).

@ <http://www.openmdx.org> (<http://www.openmdx.org>)