

# **----- xinco DMS -----**

**Document and Information Management System**

## **User and Administration Guide**

**29. Juni 2010 June 29, 2010**

History		
What?	When?	xincó DMS Version?
Initial Release	May 21, 2005	1.9.0
Revision	January 15, 2006	1.10.0
Revision	May 19, 2006	1.11.00
Revision	January 24, 2007	2.00.00
Revision	February 18, 2008	2.00.03
Revision	July 30, 2009	2.00.05
<u>Revision</u>	<u>June 29, 2010</u>	<u>02/01/00</u>

All product or company names mentioned in this publication might be trademarks or registered trademarks of their respective owners without being explicitly marked as such.

1. Introduction.....	6
1.1. Document and Information Management.....	6
1.2. xincos DMS – What’s special about it?.....	6
1.3. Terms and Definitions.....	7
2. Using xincos DMS.....	9
2.1. XincosExplorer.....	9
2.2. Connecting to a Server.....	9
2.3. Repository.....	11
2.4. Adding Folders and Data.....	11
2.5. Access Rights.....	13
2.6. Logging.....	15
2.7. General Data Characteristics and Operations.....	16
2.7.1. Attributes.....	16
2.7.2. Cut and Paste.....	16
2.7.3. Publish Data.....	16
2.7.4. Lock Data.....	18
2.7.5. Remove Data.....	19
2.8. Files.....	19
2.8.1. Download or Preview.....	19
2.8.2. Checkout / Checkin.....	20
2.8.3. Keeping Revisions Stored.....	20
2.8.4. Archiving.....	20
2.9. Text.....	21
2.10. URLs.....	21
2.11. Contacts.....	23
2.12. Searching the Repository.....	23
2.13. Audit Trail Feature.....	25
2.14. New Password Features.....	26
3. xincos DMS Administration.....	27
3.1. XincosAdmin.....	27
3.1.1. User and Groups.....	28
3.1.2. Languages.....	30
3.1.3. Data Types and Attributes.....	30
3.1.4. Empty Trash – Permanent Data Removal.....	30

3.1.5. Rebuilding the Index.....	31
3.1.6. Audit Trail Menu.....	31
3.2. XingoCron.....	33
4. Advanced xingo DMS Administration.....	34
4.1. Security.....	34
5. xingo DMS Development.....	35
5.1. Contributing File Indexers.....	35
5.2. Translations.....	36
5.3. Data Types.....	36
5.4. Integration with OpenOffice.....	36
5.4.1. Starting OpenOffice as a Service.....	36
5.4.1.1. Creating an OpenOffice service on Windows.....	37
5.4.1.2. Creating an OpenOffice service on Unix-like systems.....	39

# 1. Introduction

## 1.1. Document and Information Management

*“Originally, a document management system was a computer program (or set of programs) used to track and store images of paper documents. More recently, the term has been used to distinguish between imaging and records management systems that specialize in paper capture and records respectively. Document management systems commonly provide check-in, check-out, storage and retrieval of electronic documents often in the form of word processor files and the like.”*

[[http://en.wikipedia.org/wiki/Document\\_management](http://en.wikipedia.org/wiki/Document_management)]

In its early stages xincos DMS was primarily developed as an “Electronic Document Management System”, but also features multi-document import and archiving functionality, that makes it suitable for managing large numbers of “Paper Documents” which are either scanned or “printed” directly into an electronic file format like PDF.

Besides being contained in files, information can come in various other flavors. xincos DMS is easily extensible and already supports so-called data types like URLs, Contacts and plain text stored directly inside the database.

## 1.2. xincos DMS – What’s special about it?

xincos DMS is a web-service-based client-server-application consisting of xincos DMS Server and a client application called XincosExplorer. These days, many DMS are solely browser-based, simplifying deployment in business environments, but also limiting the way people are used to working with files and documents.

xincos DMS tries to offer both advantages: its platform-independent client application XincosExplorer is deployed and auto-updated via the web (Java WebStart / Java Network Launching Protocol, JNLP), its server application’s open interface can be accessed by XincosExplorer and any third-party application via “Simple Object Access Protocol” (SOAP) and “SOAP with Attachments” and XincosPublisher still offers a way to access published data with a common web browser, without having to install any additional software at all.

That way, users working with xincos DMS every day will appreciate the complete set of features and ease-of-use of XincoExplorer, while occasional/external users or customers are not at all excluded from accessing the data actually stored.

Also it is worth mentioning, that xincos DMS' client and server applications are basically independent. This means that you can set up various xincos DMS Servers on your intranet / the internet and any user with XincoExplorer installed once can access any of these servers: xincos DMS is like FTP, but with all the powerful features of a Document Management System (DMS)!

### 1.3. Terms and Definitions

General	
DMS	Document Management System
J2SE	Java 2 Standard Edition, the core components of the Java platform
J2EE	Java 2 Enterprise Edition, server-side extension to the J2SE, applications do not run standalone, but run inside a J2EE container that controls their execution
J2SE Runtime Environment (JRE)	Basic environment for running Java applications. <b>Required to run XincoExplorer. (1.5 and above)</b>
J2SE (Software) Development Kit (JDK / SDK)	Includes a JRE + compiler and tools to create Java applications. No additional JRE necessary, when JDK is present. <b>Required to run xincos DMS Server</b> because Web Application Servers compile components at runtime!
Java Application Server	Container executing J2EE applications, supporting the full J2EE stack, e.g. jBoss AS, BEA WebLogic, IBM Websphere
Web Application Server	Container executing J2EE web applications, supporting only the web portion of J2EE (JSP, Servlets, etc.), e.g. Apache Tomcat
SOAP Messages	Simple Object Access Protocol,

	messages or Remote Procedure Calls (RPC) expressed in XML and sent over HTTP
Web-Service	Functionality provided by a server application, consumed by a client application, client uses SOAP to access functionality and receive results created on the server-side
Java Web Start	Sun's implementation of the Java Network Launching Protocol (JNLP) to download and auto-update Java applications over HTTP and run them in a Sand Box on the local machine.
Sand Box	Container running applications isolated and with limited access to the local machine, e.g. Java Web Start applications only get full access to local drives and networks after being signed by the publisher and trusted by the user!

<b>Specific to Document Management and xincos DMS</b>	
Server / Service-Endpoint	Internet/Intranet address of the web service provided by a xincos DMS Server
Repository	All data stored on one xincos DMS Server
Meta Data	Additional information describing an object, e.g. keywords describing the content of a file.
ACL	Access Control List, consists of one or more ACEs
ACE	Access Control Entry



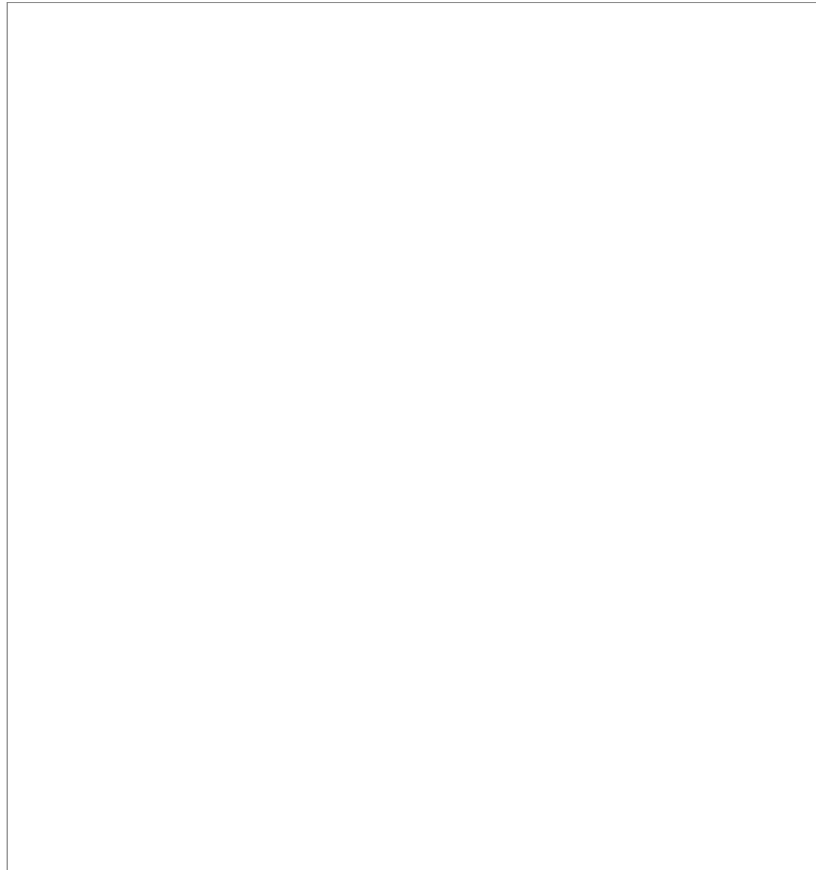
## 2. Using xingo DMS

### 2.1. *XingoExplorer*

XingoExplorer is the Java client application of xingo DMS and the standard application supporting all features of xingo DMS.

There are two ways to start or install XingoExplorer:

- Launch XingoExplorer from a xingo DMS Server of your choice with Java Web Start. When connected to the internet, Java Web Start can automatically update the application from the server it was originally loaded from.
- Download XingoExplorer, unzip the application and launch it with one of the startup scripts.



### 2.2. *Connecting to a Server*

XingoExplorer is comparable to a FTP client application, it can connect to any xingo DMS Server and access its repository, given that the user has an account on it.

Different connection profiles can be saved. They consist of a Profile Name, a Server Endpoint, a Username and Password, as well as the option of saving the password on the local computer.

A Server Endpoint is the internet/intranet address of the web service provided by the target xincos DMS Server. Web Services work on top of the Hypertext Transport Protocol (HTTP) commonly used when browsing the internet, but work on top its secure counterpart HTTPS as well. When you connect to a Server Endpoint starting with “https://...”, all data will be transported encrypted. This is the preferred way of running xincos DMS on the internet, so you should ask the system’s administrator whether this option is provided.



After successfully connecting to a server, some information about this target system is displayed. Among this information you see the version number of the target server. Be aware that although xincos DMS developers try to preserve compatibility between servers and clients of different versions, you should load an instance of XincosExplorer directly from your target server if you run into issues or observe unexpected behavior.

Note: A progress bar have been added to avoid the “application frozen feeling” from lengthy procedures like login, download and upload files.

Because xincos DMS is Open Source, it might even be possible that accessing a specific server requires you to load an extended/modified/customized version of

XincoExplorer from that server. Administrators should point this out on the starting page of their server!

### 2.3.Repository

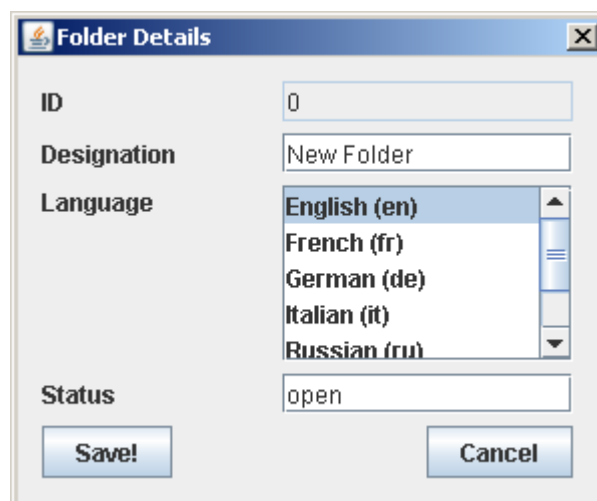
The view of the repository, stored on a xinco DMS Server, consist of two parts:

- A tree structure displaying the hierarchy of folders and data (left)
- A table displaying details about the currently selected object (right)

**IMPORTANT:** A xinco DMS Repository is a strict hierarchy! Every object has exactly one location and exactly one parent object! There is no linking of one data object to many folders! To access a folder or data object, the user needs read access from the root, through all levels of folders to the desired object!

### 2.4.Adding Folders and Data

You can add sub-folders or data to any folder that you have write access to.



ID	0
Designation	New Folder
Language	English (en) French (fr) German (de) Italian (it) Russian (ru)
Status	open

Save! Cancel

Folders only consist of a designation and a language attribute, describing their content. The language selection for folders and data always contains a “Not Specified” element as well. Pre-selected is the language of the user interface.

This information can be modified any time.

Adding data objects to the repository in general is more complex and requires several steps, but the complexity of options is mainly determined by the type of data.



xincos DMS natively supports the following data types:

- Contacts consist of personal and address information stored directly inside the server's database. You can open your default email application directly from XincosExplorer with a new email message to the selected Contact.
- Files are stored on the server, their meta data is stored directly inside the server's data base, the actual file inside a protected directory of the server's file system (keeping the database small and fast). Files can be checked out for editing, checked back in to make changes available to others, new versions can overwrite old ones or be stored as a separate revision. Files can be archived (= exported) automatically to keep the active repository small and handy.
- Text is stored directly inside the server's database, is easy to edit and perfectly suitable for storing short notes or any kind of unformatted textual information.
- URLs link to websites or external resources and complete the requirements for building a universal repository that does not exclude certain types of information. URLs can be opened directly from XincosExplorer with the local system's default internet browser.

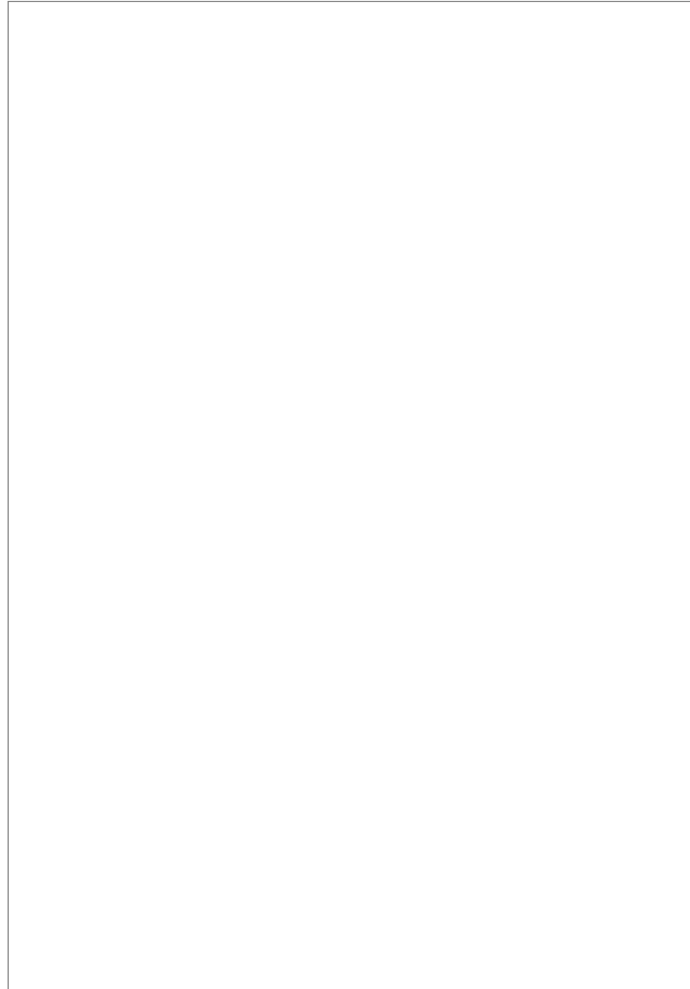


Similar to folders, basic data details only consist of a designation, the language attribute and a certain status, that is modified by various operations. In addition to these basic settings, each data type has more specific ones.

## **2.5. Access Rights**

Folders and data objects are associated with certain access rights. The basic access rights include:

- Read, required to access the content of a folder or a data object
- Write, required to add objects to a folder or modify data
- Admin, required to modify the Access Control List (ACL) of a folder or data object
- Execute, reserved for execution of server-side operations onto a folder or data object



An object's ACL consists of full access rights for a specific user, the owner or creator of the object, and additional access rights for the system's groups. Access rights cannot be granted to individual users, mainly because a xincos DMS Server is not capable (on purpose! for performance and security reasons!) of exporting a list of all registered users to a client.

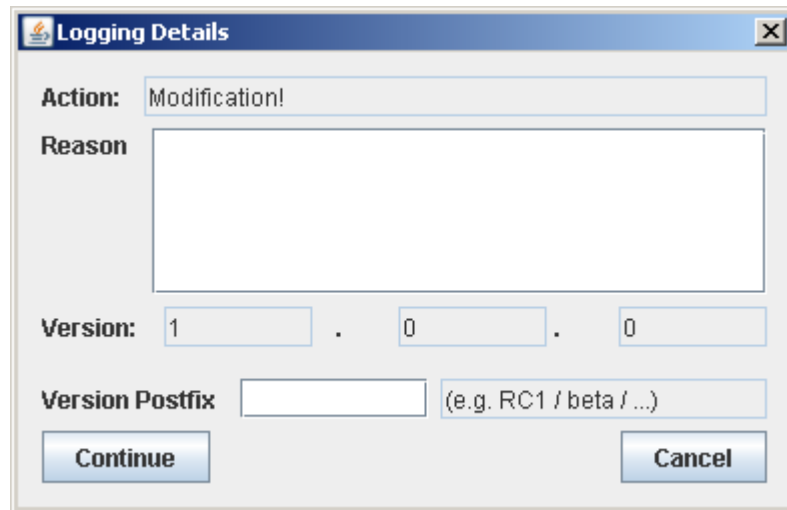
**IMPORTANT:** Users only get the access rights granted to the groups they belong to or granted to them individually. Access rights are additive, meaning the right is granted if at least one of the matching ACEs grants this right.

**IMPORTANT:** Within the strict hierarchy of a xincos DMS Repository, effective access rights to a specific object are determined from the root of the repository, through all levels of folders to the specific object.

**EXAMPLE:** An object's ACL might grant full access to the object itself, but it is sufficient to remove read access from one of the parent folders to prevent others from "finding" and accessing the object.

**IMPORTANT:** When adding new folders or data, each new object receives full access rights for its owner / creator and inherits all group access rights from its parent folder!

## 2.6.Logging



Every single modification to a data object or operation executed on it is logged. Logging Details include the type of action, a user comment and the version number associated with the new state of the object.

Version numbers reflect the fairly universal versioning model commonly used in software development:

- Major Revision
- Minor Revision
- Bugfix
- Postfix, describing a special state, e.g. “Release Candidate 1 (RC1)” , “BETA”

Logging is also used when commenting data, for example to give instructions how an object should be modified by another editor.

In order to comply with 21 CFR part 11 the reason field have been added to any action modifying an existing record in the database. This information is logged in the database along with the person making the change and the time it took place, keeping an audit trail of changes for each document. See the audit trail section for more information.

## **2.7. General Data Characteristics and Operations**

### **2.7.1. Attributes**

In addition to the basic data details and special attributes associated with certain data types, administrators can define additional attributes on their server. A list of these attributes is displayed as a table and a value can be assigned to each one of them. The screenshot shows the default attributes defined for files. When more attributes get specified for files by an administrator, they are dynamically displayed for editing in the table above.

Specific attributes of the various data types will be discussed later.

### **2.7.2. Cut and Paste**

Folders and data objects are moved within the repository by “cutting” them to the clipboard and “pasting” them into the new location. These functions are available by context menu, main menu or keyboard shortcuts “Ctrl + X” / “Ctrl + V”.

### **2.7.3. Publish Data**

While XincosExplorer is a full-featured client application for xincos DMS Servers, capable of accessing, modifying and managing the repository, it is often necessary to make data available to the general public without having them use a special application. This is when XincosPublisher is used as a light-weight “client” for read-only access to the repository.

There are 2 ways of “publishing” data:

- Setting its status to “Published”
- Granting read access for group “Public”

#### ***Setting status to “Published”***

Data can be published from within XincosExplorer, the action is logged with an appropriate logging comment and the status of the object is set to “Published”.

Two characteristics come with this status:

- The object cannot be modified anymore.
- While the object is still bound to its existing ACL within the repository, it can also be publicly accessed read-only by using XincosPublisher.

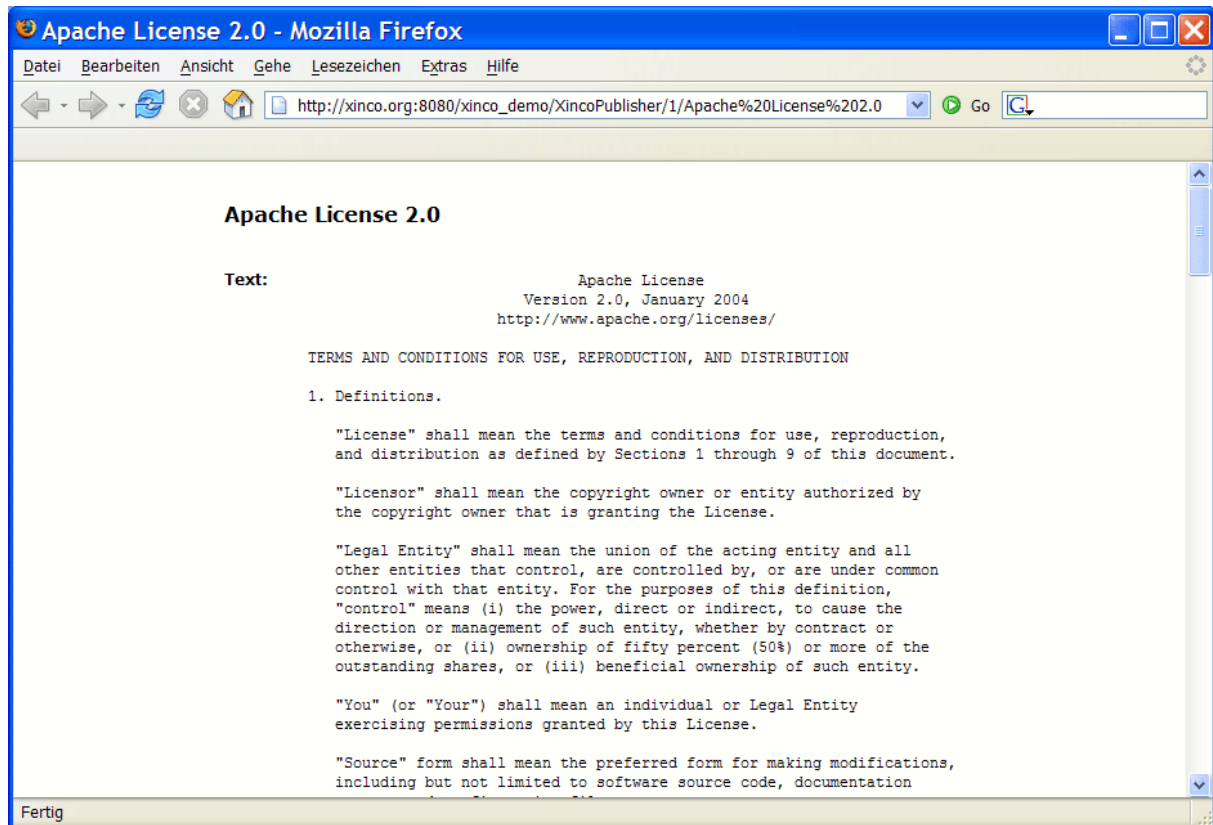
#### ***Granting read access for group “Public”***



By simply adding read access for the group “Public” to folders (!!!) and data, they can be made accessible through XincosPublisher while still being editable, not finalized.



In detail, XincosPublisher is a servlet that is part of the xincos DMS Server web application and can be accessed with any web browser. If called without parameters, it displays the following information screen, explaining its usage.



When XincosPublisher is called with the appropriate parameters,...

- ...the mandatory ID of the desired data object...
- ...plus its (path +) designation or file name (optional)...

...the content of the data object is displayed inside the browser window (Text, Contacts, etc.) or the requested file is delivered for download.

Note that before XincosPublisher delivers any content, it checks whether the requested data's status actually is "Published" or read access is granted for the group "Public".

XincosPublisher is great for linking documents from websites or intranet sites. The documents can be managed and stored inside the repository, while portals or websites reference the documents with links to XincosPublisher. There is no need to store copies of the documents on a second internet/intranet server. Store documents in one location, but reference them from anywhere you want.

#### 2.7.4.Lock Data

Locking data objects prevents them from being modified. When locking the object, a comment can be added for explanation. This is a good way to keep data stored, but at the same time tell others not to work with the data anymore or use an alternative data object instead.

Locked data can still be published and vice versa, but not made editable again.

### 2.7.5.Remove Data

With XincosExplorer, there is no way to permanently remove folders and data from the repository. Instead, the following process is required to delete data:

- Move folders or data to the special “Trash” folder.
- An administrator is supposed to review the objects in that folder every once in a while. If removal seems ok, the administrator needs to log into XincosAdmin and select “Empty Trash”. The data / files will then be permanently removed from the database and disk.

This process is best to help users maintain their part of the repository and still have the small group of administrators control critical permanent file removal.

**IMPORTANT:** All users can move their folders and data to the “Trash” folder, but once they refresh their view of the repository or when other users access the “Trash” folder, no content is displayed at all! Administrators can review the content of that folder though.

## 2.8.Files

Files are a special data type in xincos DMS because they are stored in two different ways. The actual file is stored within a protected directory of the file system, that only the server application should have access to. All meta data, attributes, ACLs or logging information referencing the file are stored inside the server’s database. Storing files that way still keeps the database small and fast, while lots of additional information can be stored with your files and documents.

The content of known file types is included into the full-text search index of xincos DMS. Also xincos DMS provides an easily extensible plug-in interface for writing custom document indexers. When missing full-text indexing support for a certain file type, contact the xincos DMS community to learn how to write your custom indexer with almost no effort.

### 2.8.1.Download or Preview

Files can be downloaded or previewed in read-only mode. When downloading a file, the user is prompted for a location to save the file at. By double-clicking on a file, it is downloaded and opened directly for preview.

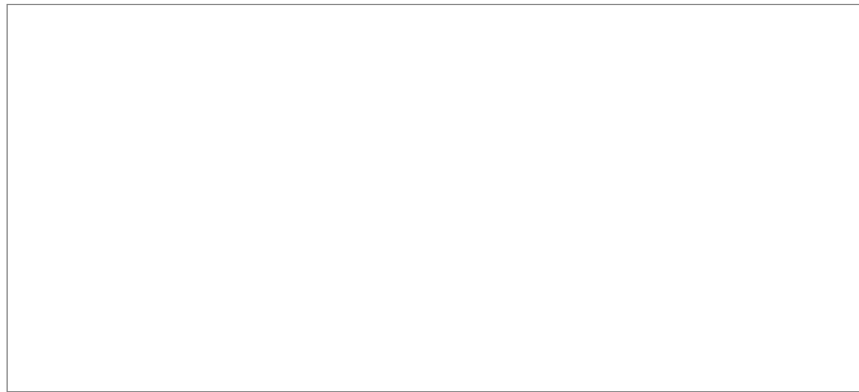
### 2.8.2. Checkout / Checkin

In order to modify files, the following process is required:

- Checkout the file; others can still download it read-only, but cannot check it out for modification themselves
- Checkin the modified file to make changes available to others.
- Undo Checkout to cancel modification and allow others to check it out.

**Note:** As of version 2.00.05 minor revisions are not available in the web application. Only major versions are available. All versions are available via Xincos Explorer at all times, assuming the user has access to the file in question.

### 2.8.3. Keeping Revisions Stored



Besides storing additional information, the user has the choice of either overwriting a file with every new checkin or storing new and old file as separate revisions. Revisions previously kept in storage can always be accessed by selecting “Download Previous Version”.

### 2.8.4. Archiving

When installing xincos DMS Server, a server-side archive directory is set for files to be exported to. When adding files to a repository, an archiving model can be selected:

- Either files are not archived at all.
- Files are archived on a fixed date.
- Files are archived x number of days after being imported into the repository.

Archiving settings can be modified any time.

xincos DMS Server runs a background service that checks for files to be archived every day. Once it detects files to be archived, it modifies their status to “Archived”, creates a sub-directory structure within the specified server-side archive directory that matches the folder structure / location of the file within the repository and exports the latest file and all previous revisions to that location on disk.



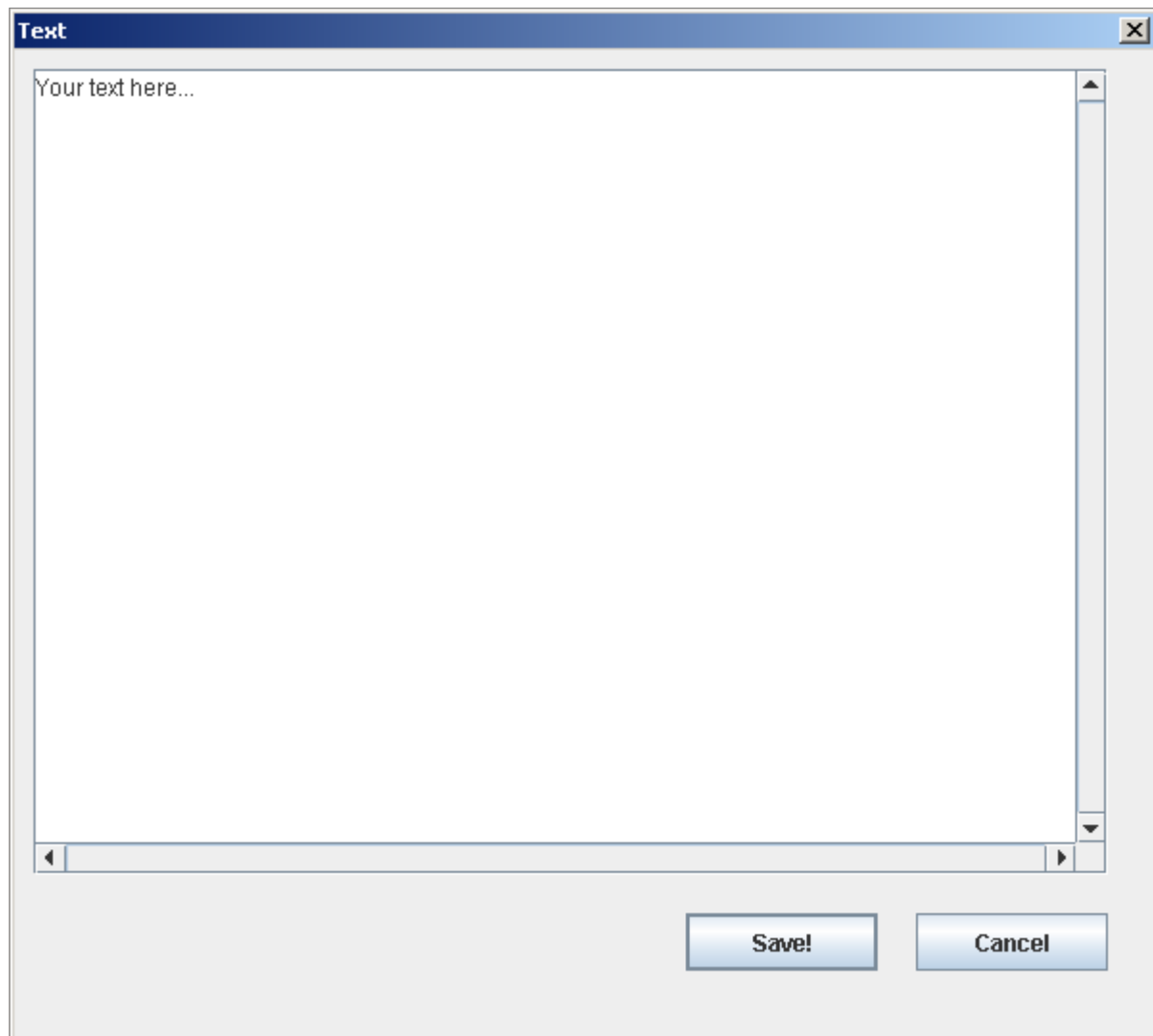
From the archiving directory, administrators can move the data to a permanent backup medium. All meta data, logging history and references to the file remain stored inside the database, while the active repository is kept small by exporting disk space consuming files.

Especially when xincos DMS is used to store documents that lose importance after a certain period of time (e.g. invoices that need to be kept for only x number of years), the archiving feature can be used to export them automatically after expiration.

## **2.9. Text**

Text is modified with a special multi-line edit window, is stored directly inside the database and is easy to modify at any time. There is no Checkout/Checkin procedure for modifying text. This data type is perfectly suitable for storing notes, license information or any kind of plain text.

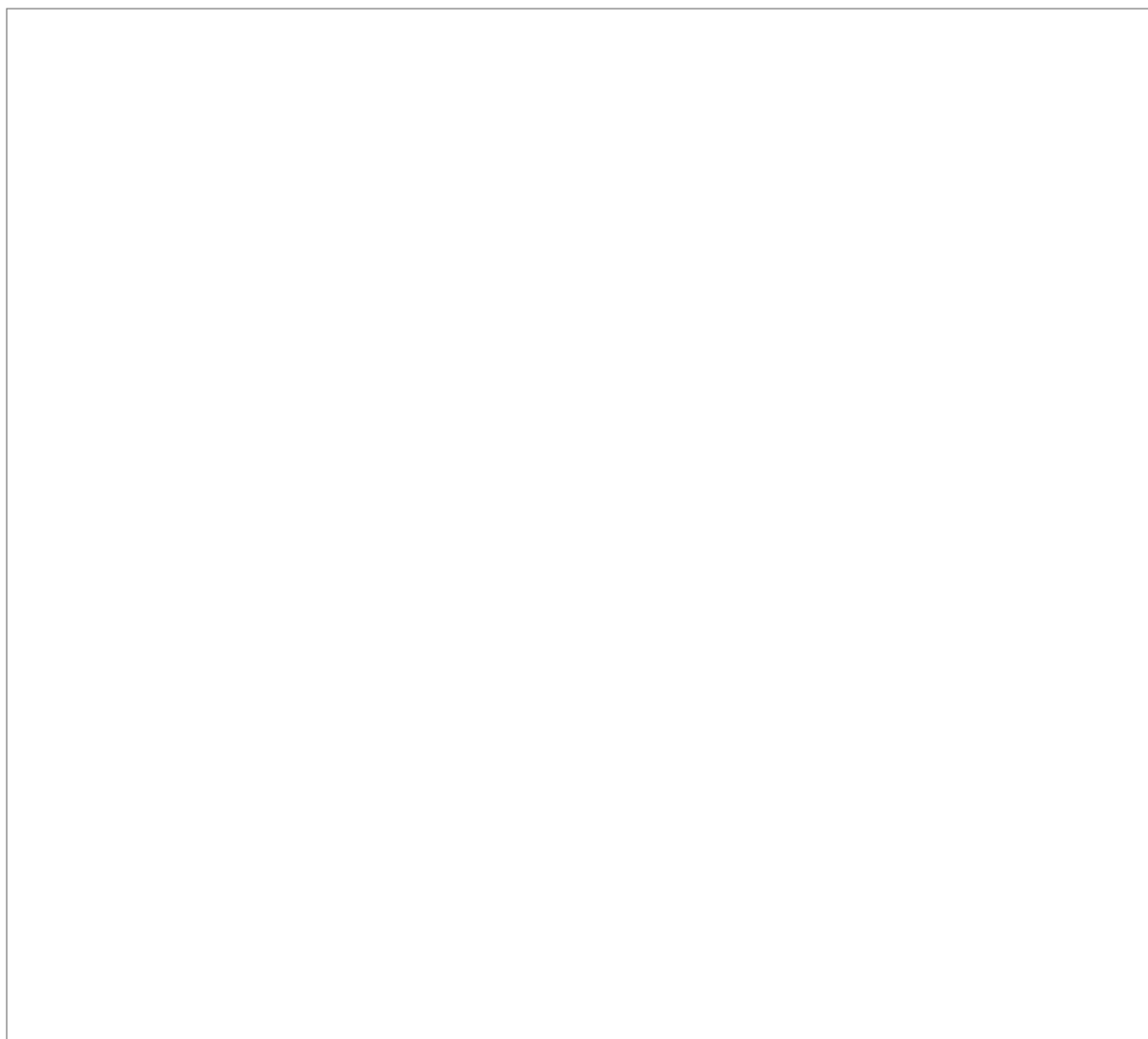
## **2.10. URLs**



The image shows a standard Windows-style dialog box titled "Text". It features a large text area with the placeholder text "Your text here...". To the right of the text area is a vertical scrollbar. At the bottom right of the dialog, there are two buttons: "Save!" and "Cancel". The dialog has a blue title bar and a close button (X) in the top right corner.

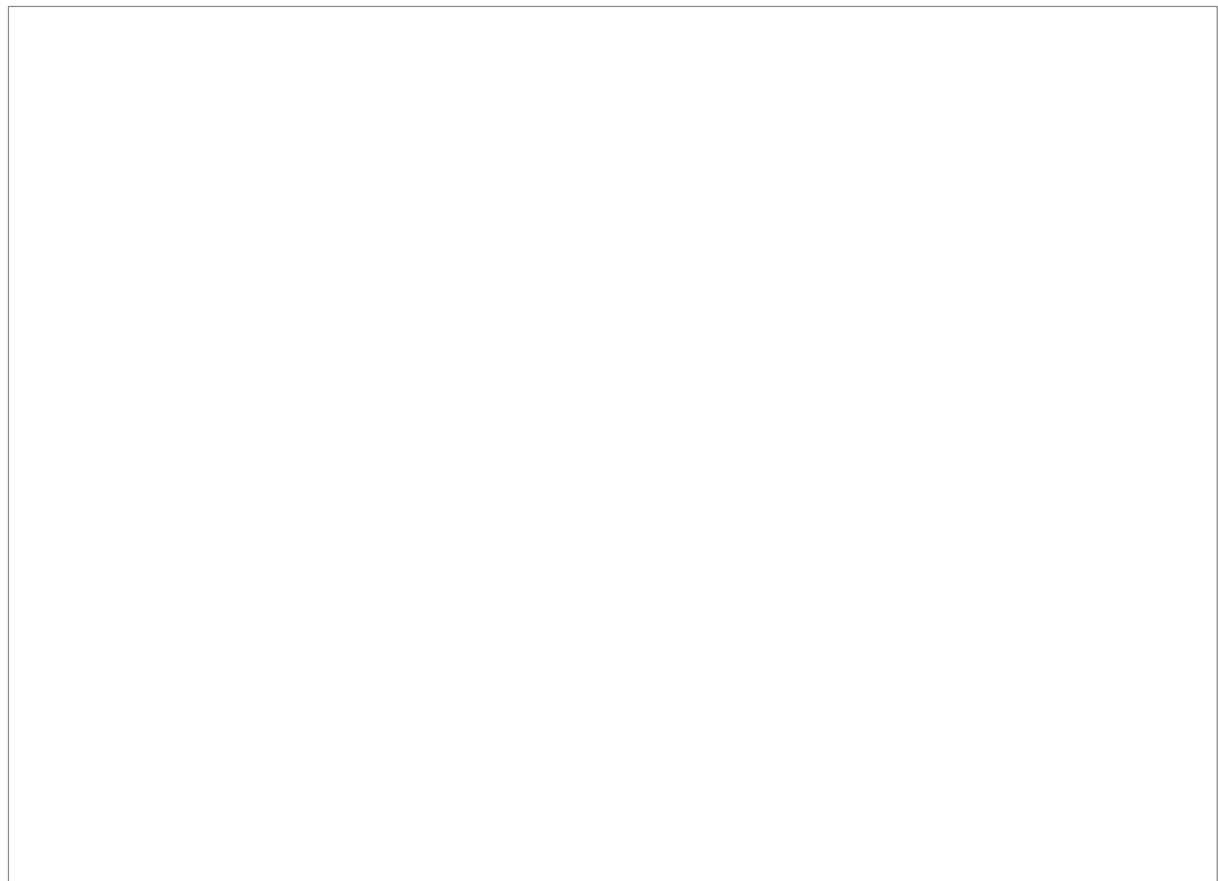
URLs are links to external resources like websites or documents on the internet or intranet. They can be opened directly with your system's default web browser.

## **2.11. Contacts**



Contacts store personal and address information. You can start a new email message to a contact by opening your default email application directly from XincoExplorer.

## **2.12. Searching the Repository**



xincos DMS provides full-text indexing for common file types like Microsoft Office documents, PDF, plain text, etc. It also indexes data designations and all additional attributes. On the server-side, Apache Lucene is used to maintain and search the index, so xincos DMS supports Lucene's full query syntax. XincosExplorer offers a simple visual search query builder listing possible operators and fields/attributes for easily and rapidly building even complex search queries. Because data objects can be associated with a certain language, this criteria can optionally be added to a search query.

The following table provides an overview of the query syntax:



<b>Using fields</b> (default field is “designation”)	
title:oneword	
title:”More than one word”	
<b>Using wildcards</b>	
tes?t	? = single character wildcard
tes* / te*t	* = multi-character wildcard
<b>Using fuzzy searches</b>	
roam~	~ = fuzzy operator
roam~0.8	~ with similarity between 0 and 1
<b>Using proximity searches</b>	
“xincos DMS”~10	2 words within 10 words of each other
<b>Using range searches</b>	
[200 TO 300]	Inclusive, between 200 and 300
{Aida TO Carmen}	Exclusive, between the 2 words
<b>Boosting a term</b>	
xincos^4 DMS	Boost xincos’s relevance over DMS
<b>Using Boolean operators</b>	
OR	
AND	
+	Require term after +
NOT	Exclude if term after NOT is in data.
-	Exclude if term after - is in data.
<b>Grouping</b>	
(xincos AND DMS) OR Java	
<b>Using field grouping</b>	
title:(+Java +”xincos DMS”)	
<b>Escaping special characters</b>	
Character to escape special characters:	\
List of special characters	+ - &&    ! ( ) { } [ ] ^ " ~ * ? : \

**IMPORTANT:** The result list includes only data objects that you have access to! Checking these access rights might take a few seconds if your search request results in many hits!

## 2.13.Audit Trail Feature

The major improvements within Xincos 2.0.0 are the addition of audit trail features to comply with 21 CFR part 11 FDA regulations. This provides the following benefits:

1. Know who, what and when something was changed within the system.
2. Get a change log for ANY record in the database.

3. With the combination of the audit log, already implemented document versioning and administrator knowledge; you should be able to move any document to any stage it has been thru in the past.
4. Have FDA ready audit trail records.

Any change to a document, including creation and/or deletion, is logged into the system. When any of those changes occur the system prompts for a reason. This reason is mandatory and can't be empty. Along this explanation for the change, the user, the original record and the time of the change is registered in the database.

## **2.14.New Password Features**

In order to comply with 21 CFR part 11 FDA regulations the passwords needed to have 2 more features that are built into Xinco Explorer and Xinco Administration and are as follows:

1. Password aging: after a certain amount of days (default 120 days) the system will prompt for a new password. The password can't be repeated in a certain period of time. (default 365 days)
2. Account lock: After a certain number of failed login attempts (default 3) the account will be locked.

## 3. xinfo DMS Administration

### 3.1.XinfoAdmin



XinfoAdmin is part of xinfo DMS' server-side management tools. Login is required to enter the main menu of the application.

**IMPORTANT:** Only members of the pre-configured Admin-Group (ID=1) can login to XinfoAdmin using their regular username and password.

**New Feature:** XinfoAdmin is now internationalized as well!

XinfoAdmin offers the following actions:

- Edit user accounts.
- Edit groups and assign users.
- Edit your own user profile.
- Edit languages
- Edit data type attributes
- Permanently remove data from "Trash".

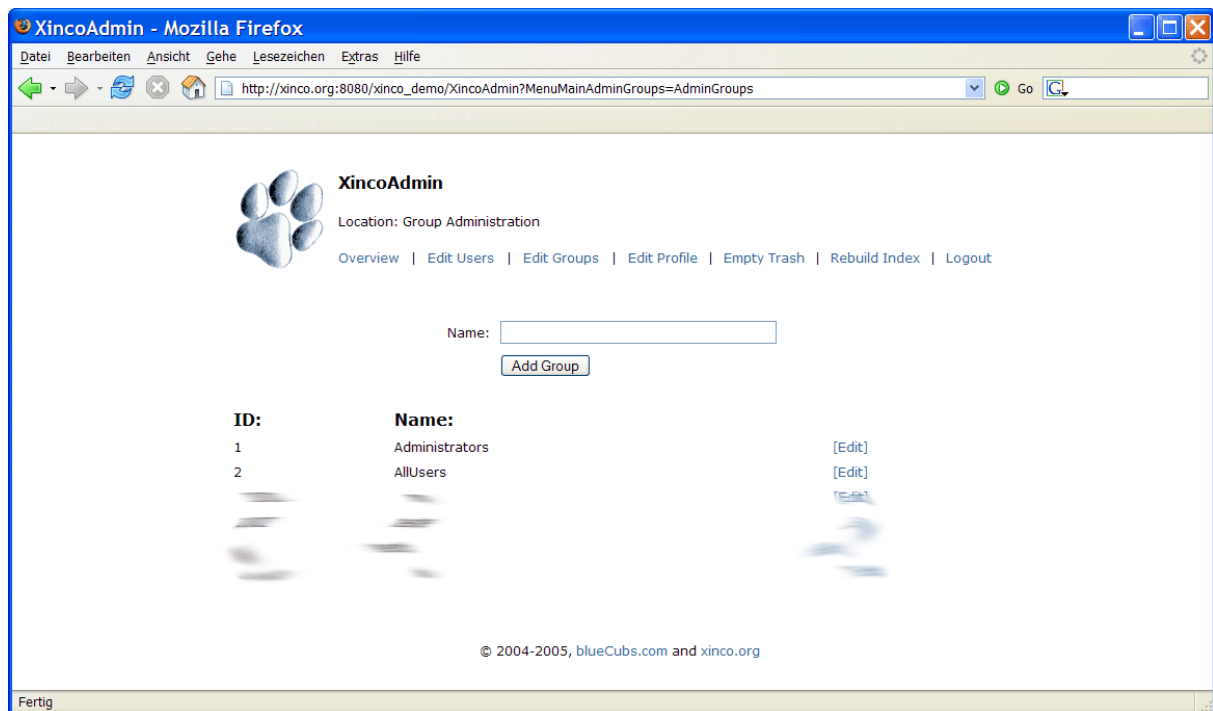


- Rebuild the search index.

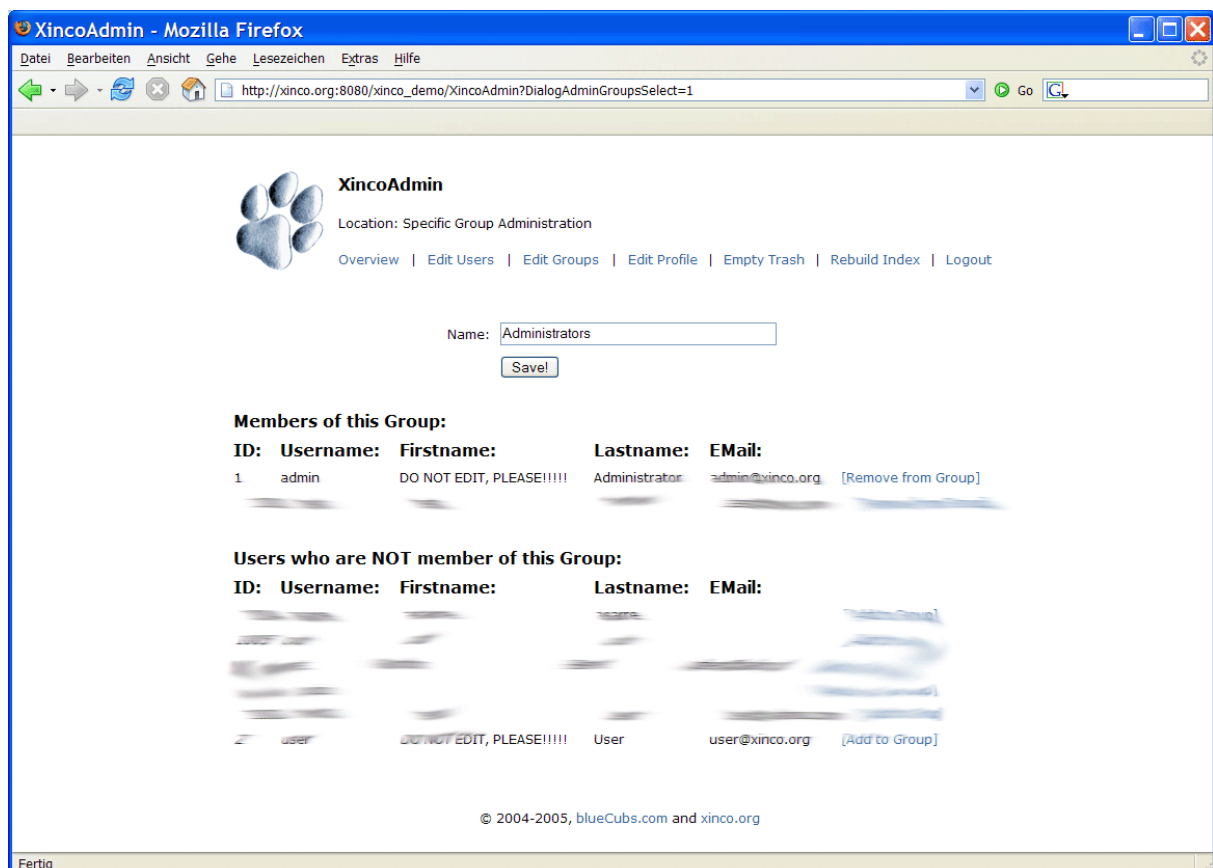
### 3.1.1. User and Groups

New users are added to xingo DMS by using XingoAdmin. It is mandatory to specify a username, a password, first- and last name, as well as a valid email address. Each user can modify personal information and the password any time using XingoExplorer. Once added, “Username” cannot be modified anymore.

**IMPORTANT:** Existing users cannot be deleted because they are referenced by logging entries. To prevent that a user can log on to the system, lock the user!



New groups can be added by specifying a name, or can be edited by selecting them.



The screen above shows the detail page of a specific group. Its name can be modified and users can be added or removed from the group, affecting their effective access rights.

### 3.1.2.Languages

Languages are supposed to describe the content of folders and data. Because people tend to store data in only a few languages they can understand, xincos DMS does not come with a list of hundreds of languages, but encourages administrators to add languages as they are required.

Unused languages can also be deleted through XincoAdmin.

### 3.1.3.Data Types and Attributes

xincos DMS ships with pre-defined data types like “File” or “Contacts”, each with certain characteristics that require an implementation on client- and/or server-side. In addition, each data type has a number of attributes assigned. Files and text have one or more fixed attributes that are required for core functionality such as archiving.

Custom attributes can be added and/or removed using XincoAdmin. When adding an attribute it is mandatory to assign a position in the list and select a data type for that attribute. Removing an attribute also deletes all of its attribute values stored with data.

**BACKGROUND:** Attributes assigned to a data type are stored inside the database table *xincos\_core\_data\_type\_attribute*. They refer to a data type and carry an attribute id that determines the order in which they are listed. When adding data of a certain type, the actual values for its attributes get stored inside the table *xincos\_add\_attribute*, referencing the data object and carrying the attribute id.

**HINT:** If you know that changes to data type attributes are necessary, it is best to complete those modifications before adding any data to the repository because *xincos\_add\_attribute* will be empty!

**IMPORTANT:** Changing attributes requires a full rebuild of the search index!

### 3.1.4.Empty Trash – Permanent Data Removal

As explained before, data to be deleted is moved to the special “Trash” folder (ID=2). This folder can be emptied, deleting all data and sub-folders contained. Offering this critical option in XincoAdmin only, gives administrators full control of permanent data removal and a chance of revising content to be deleted prior to removal. Viewing the content of “Trash” is done with XincoExplorer, XincoAdmin does not show the content of this folder.

### 3.1.5.Rebuilding the Index

During installation, a directory is specified for storing a full-text index of all data. Although this index is maintained automatically, there are a few occasions that require manually triggering a rebuild of the index.

- After initial installation to create the index
- After manually modifying data types' attribute names, e.g. in database table: *xincO\_core\_data\_type\_attribute*
- After executing a database translation script
- After running update scripts, that involve renaming data types' attributes.
- When part of the index gets deleted or corrupted (system crash, etc.).

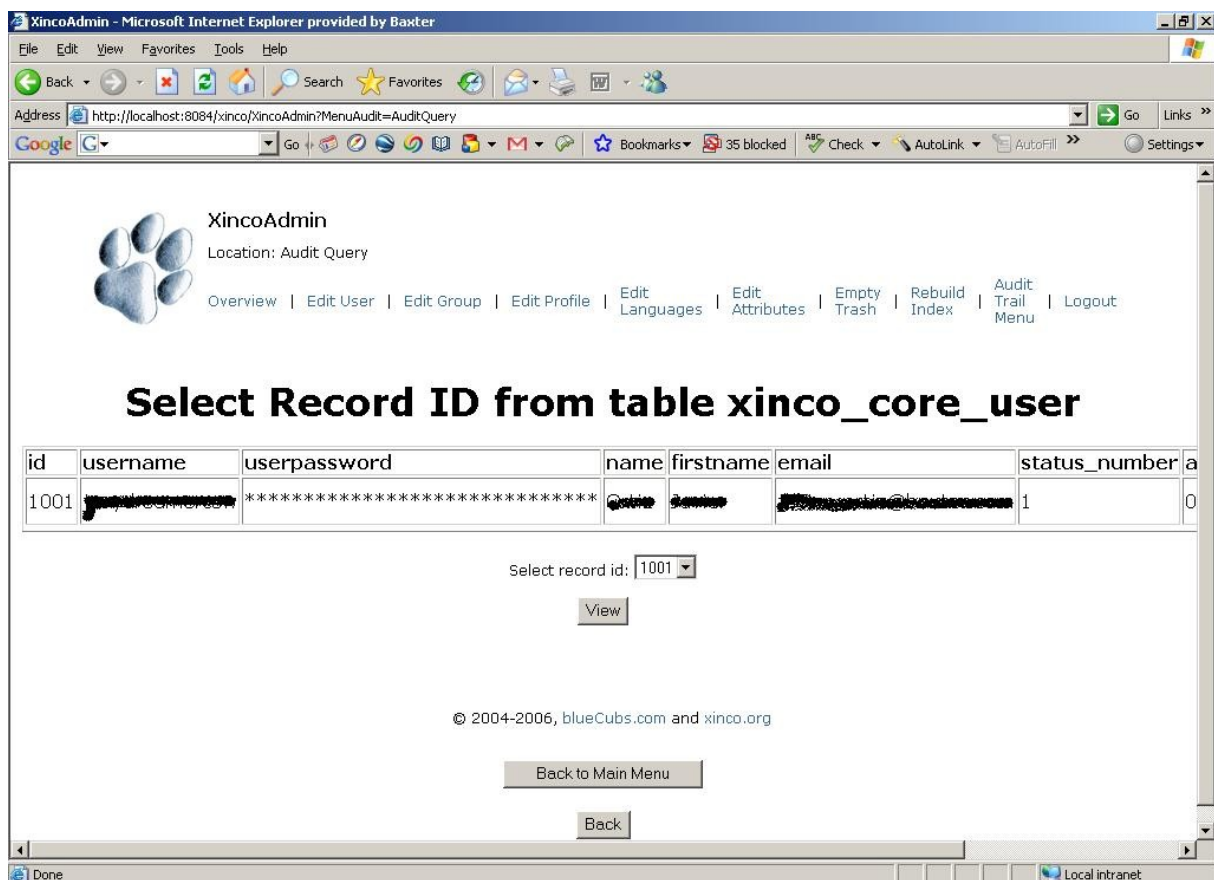
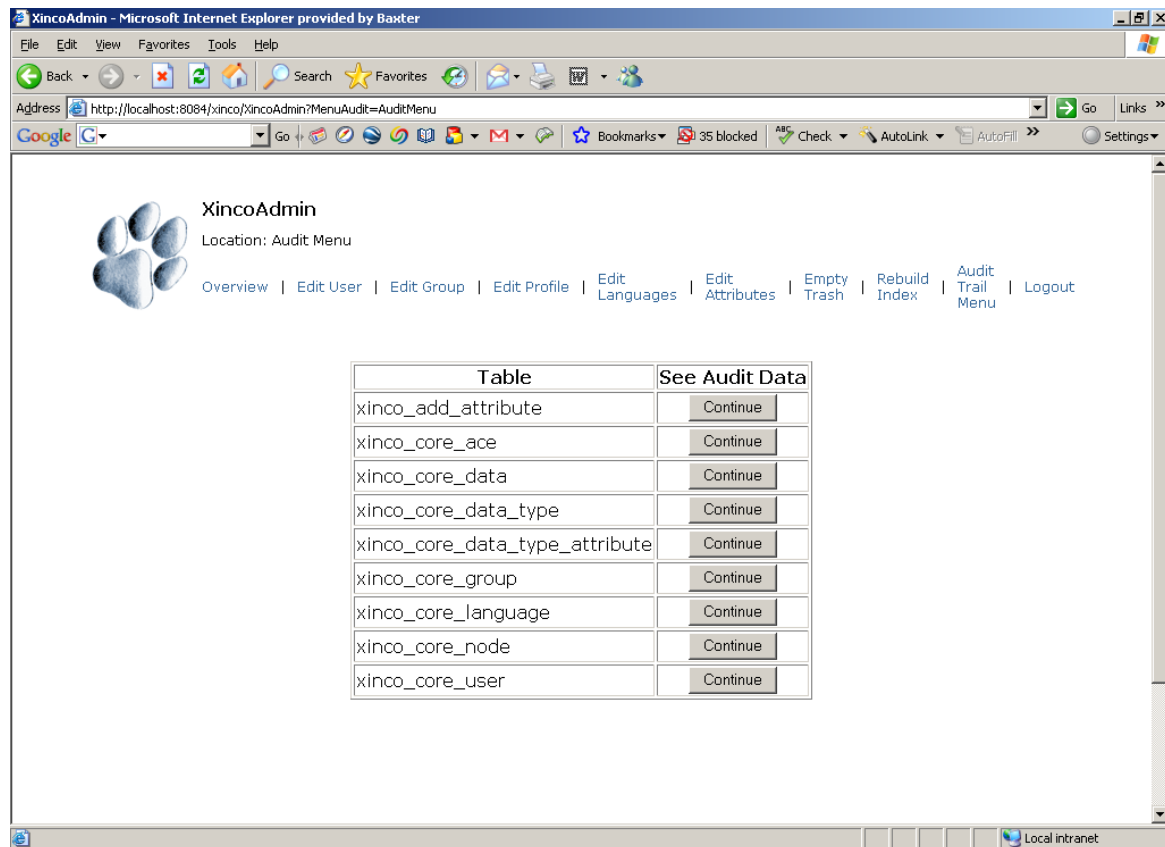
**IMPORTANT:** Rebuilding the index can take quite a while and consume lots of memory. It is not recommended to trigger full rebuilds while users can access the system!

**IMPORTANT:** Independent of manual rebuilds of the index, it is periodically optimized by a background service, so rebuilds are not required for optimization.

### 3.1.6.Audit Trail Menu

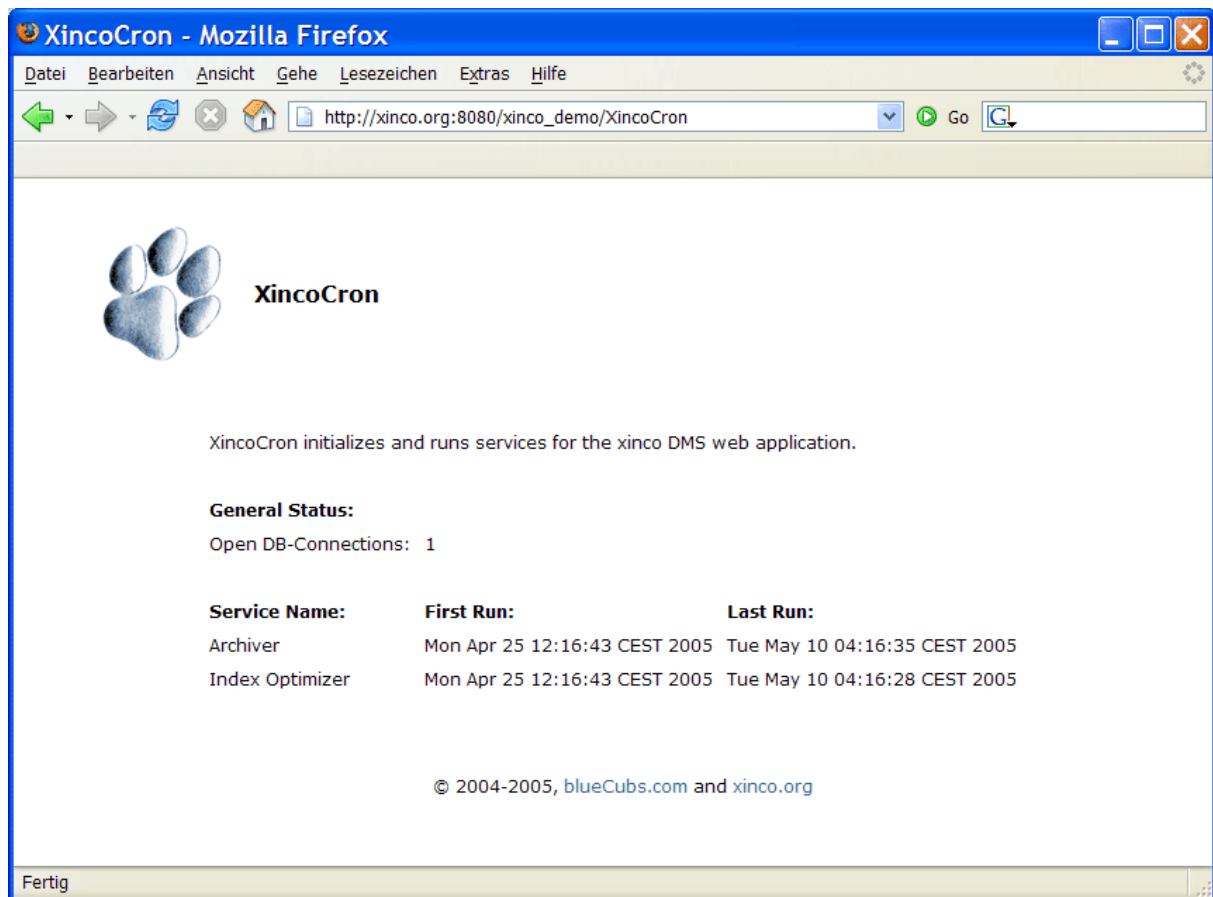
This new feature lets anyone with administration rights to see the audit trail of certain record within the database with the following features:

1. User passwords are replaced by '\*' characters for confidentiality (besides the already in place database encryption).
2. Audit trail shown in chronological order (latest at top) along with user and time of change.
3. Only records that have been modified are available for selection.
4. Action of change is internationalized.





### 3.2.XincroCron



XincroCron runs a number of background services helping to maintain a xincro DMS repository. Also it displays status information about these services and the server application in general.

The background services include:

- Archiver, periodically checking for data to be archived and exporting them to the specified location on disk.
- Index Optimizer, periodically optimizes the search index to increase performance and stability of full-text searches.

## 4. Advanced xincos DMS Administration

xincos DMS is easily customizable in many ways, a few approaches are explained in this chapter.

### 4.1. Security

xincos DMS uses SOAP over HTTP for communication between client and server applications, which implies that all requests, login information and data goes through the wire without using any security features like encryption or authentication of client and server.

To improve security, SOAP web-services work on the secure protocol HTTPS as well. Using xincos DMS over HTTPS requires a few simple steps described in external documents:

- Obtaining a SSL certificate from a certification authority or creating a self-signed certificate (e.g. for testing)
- Preparing your Web Application Server (e.g. Tomcat) for HTTPS
- (importing your self-signed certificate on the client side; not required for official certificates issued by a certification authority)
- Modifying your target server's server endpoint in XincosExplorer to use HTTPS for communication

The following resources explain how to obtain/create certificates and how to prepare Tomcat for HTTPS:

- <http://jakarta.apache.org/tomcat/tomcat-5.0-doc/ssl-howto.html>
- <http://www.pankaj-k.net/WSOverSSL/WSOverSSL-HOWTO.html>

**IMPORTANT:** When using self-signed certificates for XincosExplorer over Java Web Start, you need to export your certificate from the keystore to a “.cer”-file and re-import it as a root certificate into Java's list of trusted certificates/sites/authorities. (Java Web Start > Edit > Properties > Security > Certificates > Import into “Cert. Auth. for Trusted Sites”)

When using the non Java Web Start version of XincosExplorer, there are several batch or shell script files for creating a local keystore for your certificates and executing XincosExplorer using this keystore.

Finally you need to specify the correct server endpoint in Xincos Explorer in order to use HTTPS for communication. When your HTTP server endpoint is something like [http://xincos.org:8080/xincos\\_demo/services/Xincos](http://xincos.org:8080/xincos_demo/services/Xincos) then your HTTPS server endpoint will look like: [https://xincos.org:8443/xincos\\_demo/services/Xincos](https://xincos.org:8443/xincos_demo/services/Xincos)

- The protocol changed from “http” to “https”
- Tomcat’s default TCP port for the secure protocol is 8443 instead of 8080

## 5. xincos DMS Development

Beside customizing your xincos DMS installation, there are many approaches for getting involved into xincos DMS development or developing extended applications based on xincos DMS.

### 5.1. Contributing File Indexers

The content of various file types are indexed by xincos DMS using a special indexer for each of them. If you are missing full-text indexing for some file type, you can write your own indexer.

To see how xincos DMS does the indexing, take a look at the source code of the plugin interface and special indexer classes of the package *com.bluecubs.xincos.index*.

Just two steps are required for getting your custom indexer running:

- Write your custom indexer, compile it and add the class file to a package like *com.bluecubs.xincos.index.filetypes*
- Add your indexer to xincos’s configuration and redeploy the web application.

Changes to xincos’s configuration are required in the web applications *context.xml* file (e.g. */webapps/xincos/META-INF/context.xml*).

- Increase the count of available indexers:
- Specify the class name and corresponding file extensions to be handled by the indexer:

```
<Environment name="xincos/FileIndexerCount" value="6"
type="java.lang.Integer" override="false"/>

<Environment name="xincos/FileIndexer_6_Class"
value="com.bluecubs.xincos.index.filetypes.XincosIndexXXX"
type="java.lang.String" override="false"/>
<Environment name="xincos/FileIndexer_6_Ext" value="xxx;xy"
type="java.lang.String" override="false"/>
```

Feel free to contribute custom indexers to the Open Source distribution of xincos DMS and contact the development team if questions arise!

## **5.2. Translations**

For each release of xincos DMS, a so-called XincosLanguageMasterPack is available. xincos DMS is primarily developed in English and German, so these languages are the base for all translations. The master language pack includes Java properties files for XincosExplorer and the SQL script for translating the database of xincos DMS. If you want to do your own translation or contribute one to the Open Source releases of xincos DMS, download the master language pack, translate the files either from English or German and recompile the application or send the files back to us for inclusion into the following release of xincos DMS.

## **5.3. Data Types**

A more complex approach of extending xincos DMS is the implantation of whole new data types for certain purposes. The design of both, database and application are pretty flexible to support this.

For more information and technical details, feel free to contact the xincos DMS development team.

## **5.4. Integration with OpenOffice**

Since version 2.01.00 Xincos can be configured to use OpenOffice as a service for file type conversion using JODConverter. In order to do this make sure to comply with the JODConverter requirements. As the time of writing the current version is 2.2.2 and works with OpenOffice 3.2.1. Keep in mind that JODConverter 2.2.2 was written in 2009 and still works with current versions of OpenOffice. As long as the API doesn't change it should work with later versions as well.

### **5.4.1. Starting OpenOffice as a Service**

JODConverter needs to connect to a running OpenOffice.org instance in order to perform the document conversions.

This is different from starting the OpenOffice.org program as you would normally do. OpenOffice.org can be configured to run as a service and listen for commands on a

TCP port; there are a few ways to accomplish this but the simplest one is to start it from the command line with the following options:

- soffice -headless -accept="socket, host=127.0.0.1, port=8100;urp" -nofirststartwizard

#### **5.4.1.1. Creating an OpenOffice service on Windows**

Creating a Windows Service lets you have an OpenOffice.org instance always running as a service and listening for connections. That's what you usually want to do on a server machine. Here is one way to create a Windows Service: use the SRVANY utility that you can download for free as part of the Resource Kit. (This was done on an XP SP2, YMMV.)

1. I assume you already installed OpenOffice.org and started at least once under the current user
2. Download and install the Windows Server 2003 Resource Kit Tools
3. Read How To Create a User-Defined Service
4. Create a service named OpenOfficeUnoServer following the instructions above; as the Application registry value use C:\Program Files\OpenOffice.org 2.0\program\soffice.exe
5. Add another value to the Parameters key named AppParameters and set it to -headless -nofirststartwizard -accept=socket,host=localhost,port=8100;urp;StartOffice.Service
6. From Control Panel / Administrative Tools / Services open the Properties for the new service and change the Log On account to be Local Service which is a more secure option than SYSTEM
7. If we stop here the service will start but not actually work. That's because the service runs as a special account and OpenOffice.org will try to show the license agreement dialog as if it was a new user instead of starting the application. To avoid this, you need to edit share\registry\data\org\openoffice\Setup.xcu (it's an XML file) inside the OpenOffice.org installation directory and replace this bit

```

<prop oor:name="ooSetupInstCompleted">
  <value>false</value>
</prop>
<prop oor:name="ooSetupShowIntro">
  <value>true</value>
</prop>

```

with this other one (replace the date with today's date; it must be later than the OOo installation time)

```

<prop oor:name="ooSetupInstCompleted" oor:type="xs:boolean">
  <value>true</value>
</prop>
<prop oor:name="LicenseAcceptDate" oor:type="xs:string">
  <value>2006-07-25T17:34:04</value>
</prop>
<prop oor:name="FirstStartWizardCompleted" oor:type="xs:boolean">
  <value>true</value>
</prop>

```

8. Start your new service (e.g. from Control Panel / Administrative Tools / Services)
9. You may need to unblock the new service at the firewall level
10. From a command prompt do

```
> netstat -anp tcp
```

this should show a listening on port 8100. Chances are that the service will only listen on the 127.0.0.1 interface; this means you can only connect from the same machine and it's good from a security point of view. However, if you want to be able to connect from other machines as well you need to change the *accept* string to "socket,host=0.0.0.0,port=8100;urp;" in the AppParameters registry value

**Note:** The resulting registry entry as mentioned above is provided for convenience in Docs\xinco-DMS-UserAdminGuide\OpenOfficeService\Windows\ and is provided without any guarantee.

#### 5.4.1.2. Creating an OpenOffice service on Unix-like systems

Creating a SysV-style init script for starting OpenOffice.org is not difficult but the details depend on your particular OS flavour/distribution.

The tricky part is that OpenOffice.org requires an X server to work. On a headless server, you can use **Xvfb** as a *dummy* display for OpenOffice.org.

However since OpenOffice.org 2.3 an X server is no longer needed; it is now possible to start OpenOffice.org in "true" headless mode simply by making sure the `DISPLAY` environment variable is not defined:

```
$ unset DISPLAY
```

```
$ soffice -headless -accept="socket,host=127.0.0.1,port=8100;urp;" -nofirststartwizard
```